VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

WEB SITE OPTIMIZATION

BAKALÁŘSKÁ PRÁCE
BACHELOR´S THESIS

AUTOR PRÁCE JIŘÍ PETRŽELKA
AUTHOR

BRNO 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

OPTIMALIZACE WEBOVÝCH STRÁNEK

WEB SITE OPTIMIZATION

BAKALÁŘSKÁ PRÁCE
BACHELOR´S THESIS

AUTOR PRÁCE
AUTHOR

JIŘÍ PETRŽELKA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. MILOŠ EYSSELT, CSc.

BRNO 2007

BSc. Project Specification/4445/2006/xpetrz10

Brno University of Technology - Faculty of Information Technology

Department of Computer Systems   Academic year 2006/2007

BSc. Project Specification

For: **Petrželka Jiří**
Branch of study: Information Technology
Title: **Web Site Optimisation**
Category: Web

Instructions for project work:

1. Research how search engines are implemented and what they offer for searchers and for web developers.
2. Examine the search engine optimisation (SEO) techniques and tools used for SEO, and W3C standards XHTML, CSS (cascading style sheets) and WAI (web accessibility initiative).
3. Transform the existing website www.hcc.cz into an XHTML 1.0 + CSS 2.0 compliant form.
4. Apply the WAI recommendations to the site mentioned. The output website will comply with the Web Content Accessibility Guidelines 1.0. Optimize the website for SEO and analyze site's traffic after optimisation.
5. Produce a clone website whose static contents will be translated into English. For dynamic contents, sample English data will be produced.
6. Consult with Dr. Richard Rider, FEC, Coventry University, UK.
7. Discuss the results and suggest possibilities for further development of the project.

Basic references:

- World Wide Web Consortium, available on http://www.w3.org/
- Web Accessibility Initiative, available on http://www.w3.org/WAI/

Detailed formal specifications can be found at http://www.fit.vutbr.cz/info/szz/

The BSc. Thesis must define its purpose, describe a current state of the art, introduce the theoretical and technical background relevant to the problems solved, and specify what parts have been used from earlier projects or have been taken over from other sources.

Each student will hand-in printed as well as electronic versions of the technical report, an electronic version of the complete program documentation, program source files, and a functional hardware prototype sample if desired. The information in electronic form will be stored on a standard medium (diskette, CD-ROM) in formats common at the FIT. In order to allow regular handling, the medium will be securely attached to the printed report.

Supervisor: **Eysselt Miloš, Ing., CSc.**, DCSY FIT BUT
Beginning of work: November 1, 2006
Date of delivery: May 15, 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
612 66 Brno, Božetěchova 2

L.S.

Zdeněk Kotásek
Associate Professor and Head of Department

p

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. **Pan**

Jméno a příjmení: **Jiří Petrželka**
Id studenta: 84202
Bytem: Černého 22, 635 00 Brno
Narozen: 09. 07. 1984, Brno
(dále jen "autor")

a

2. **Vysoké učení technické v Brně**

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

....
(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
   bakalářská práce

Název VŠKP: Web Site Optimisation
Vedoucí/školitel VŠKP: Eysselt Miloš, Ing., CSc.
Ústav: Ústav počítačových systémů
Datum obhajoby VŠKP: ....

VŠKP odevzdal autor nabyvateli v:

tištěné formě	počet exemplářů: 1
elektronické formě	počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
   ☒ ihned po uzavření této smlouvy
   □ 1 rok po uzavření této smlouvy
   □ 3 roky po uzavření této smlouvy
   □ 5 let po uzavření této smlouvy
   □ 10 let po uzavření této smlouvy
   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: ....

....

Nabyvatel Autor

Abstract

The goal of this project is to enhance the accessibility and usability of an existing company presentation located at http://www.hcc.cz, boost the site’s traffic and so increase the company’s revenues.

The project follows these steps to accomplish this:

a) A partial refactoring of the back-end (PHP scripts).

b) Transformation of the website contents according to the recommendations of the World Wide Web consortium (W3C) and in particular to those of the Web Accessibility Initiative (WAI).

c) Application of the Search Engine Optimization (SEO) techniques and analysis of their impact. In this step, the project touches upon the Search Engine Marketing (SEM).

Keywords

optimization, World Wide Web, web sites, Internet, W3C, W3C standards, WAI, WCAG, SEO, SEM, HTML, XHTML, CSS, refactoring, UML, use case diagram, MVC, class diagram, search engine, search engine optimization, search engine marketing, keywords, pagerank, s-rank, back links, URL, URL rewriting, Apache, PHP, mod_rewrite, httpd.conf, Google, Google Analytics, sitemap, XML schema, DMOZ, Seznam, copywriting, bounce rate

Abstrakt

Cílem tohoto projektu je zlepšení přístupnosti a použitelnosti existující firemní prezentace umístěné na adrese http://www.hcc.cz, dále pak zvýšení návštěvnosti stránek a zisku, který stránky firmě generují.

Tato práce sestává z následujících částí:

a) Částečná refaktorizace back-endu (PHP skriptů).

b) Transformace stránek podle doporučení W3C (World Wide Web Consortium), a to zejména podle doporučení WAI (Web Accessibility Initiative).

c) Aplikace technik SEO (Search Engine Optimization). Při následné analýze důsledků SEO se práce částečně dotýká i SEM (Search Engine Marketing).

Klíčová slova

optimalizace, web, webové stránky, Internet, W3C, W3C standardy, WAI, WCAG, SEO, SEM, HTML, XHTML, CSS, refaktorizace, UML, diagram případů užití, MVC, diagram tříd, vyhledávač, optimalizace pro vyhledávače, klíčová slova, pagerank, s-rank, zpětné odkazy, URL, přepisování adres, přepisování URL, Apache, PHP, mod_rewrite, httpd.conf, Google, Google Analytics, mapa stránek, schéma XML, DMOZ, Seznam, copywriting, bounce rate

Citation

Jiří Petrželka: Web Site Optimization, Bachelor’s thesis, Brno, FIT BUT in Brno, 2007

Jiří Petrželka: Optimalizace webových stránek, bakalářská práce, Brno, FIT VUT v Brně, 2007

Declaration

The work described in this report is the result of my own investigations. All sections of the text and results that have been obtained from other work are fully referenced.

Signed:

Date:

Acknowledgements and dedications

I would like to thank all those who supported me in the course of work on this project, which primarily includes my family.

I also thank Richard Rider for allowing me to take up this particular project, based on my own choice and interests.

And finally, I value those Google people who developed Google Analytics and allowed people to use it free of charge.

Table of contents

Introduction

The websites as means of communicating information have been gaining popularity ever since the first website was created in 1991 (CERN 2005). What started as a simple combination of URL, HTTP and HTML has developed in the course of the 1990s and the following years into a collection of a range of technologies that work together.

Nowadays, websites can be used to develop large information systems with the aim to simplify various administrative tasks, improve communication and sell products online. It is obvious that the larger an information system is the more crucial it is for the system to be easily maintainable and updatable.

This requirement is usually solved by separating the application logic into layers. As for the server side, this is often done by adopting the MVC (Model-View-Controller) pattern.

Also, as websites are becoming more user-friendly and the client side exploits various technologies that work together, there is an increasing demand for the client side to be easily updatable. Again, this requirement can be satisfied by dividing the client side output into layers.

Once the website's architecture allows us to make modifications more easily, there arises another issue and that is to draw people to the website. One way to achieve this is to follow the World Wide Web Consortium (W3C) standards that provide guidance on how to make the website accessible for most clients.

However, obeying only the W3C standards will probably not be sufficient if the website's goal is to earn money. Since many people use search engines to find new websites, it is also necessary to bear in mind what they (both search engines and searchers) expect from websites, in terms of structure, contents and presentation. To reflect these expectations, Search Engine Optimization (SEO) has to be applied.

The purpose of this project is to reform an existing website so that it conforms to all the ideas stated above.

1. Back-end: Code refactoring

1.1. Analysis of the old website

1.1.1. Introduction to the website

The HC Compact is a company selling fitness equipment, dietary supplements and several other products, such as swings for children and garden furniture.

The website allows people to browse information about products that are sorted in a tree structure of categories. The user can add goods to a shopping basket and subsequently purchase them. Each product can be marked as a special offer, in which case the goods is discounted and displayed on the title page. The user can browse the products according to its manufacturer, according to its category or use a search facility.

Apart from browsing goods, the website also encompasses pages with news, important information, a page with customer queries and some information about the company itself. The user can also register and log in. A user that is logged in can submit orders more easily and track the progress of orders already placed.

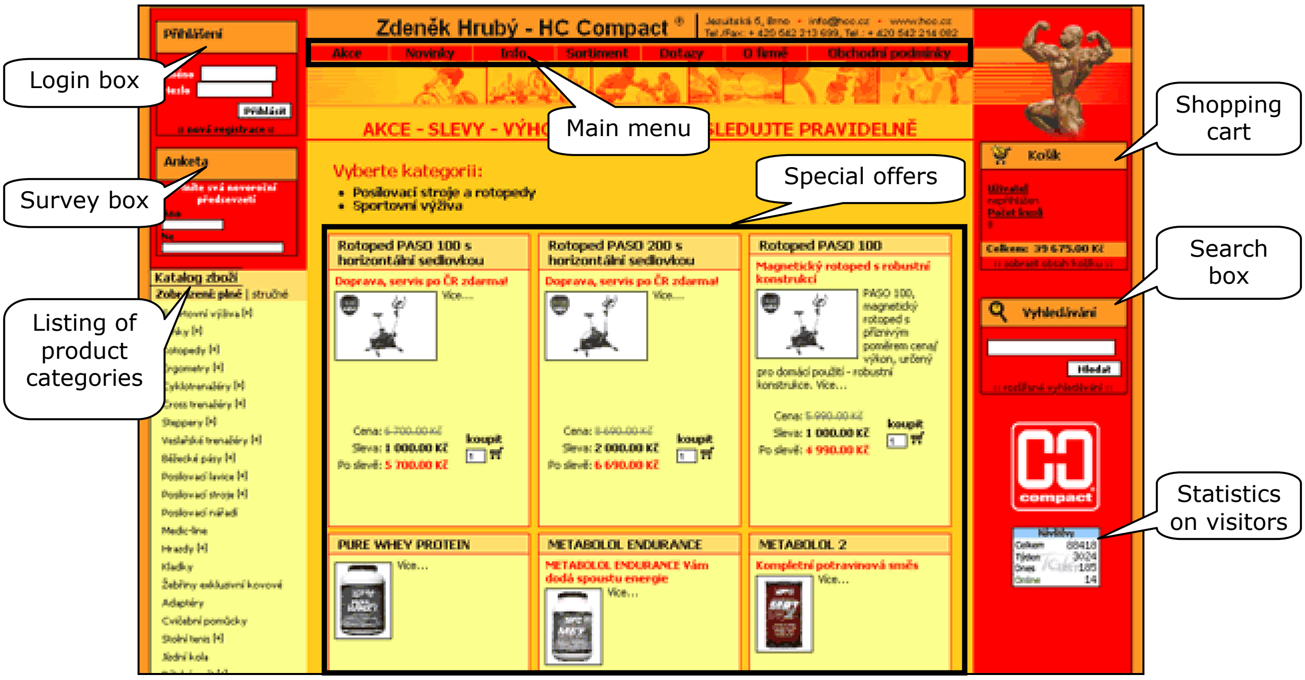


Figure 1.1.1. The home page before optimization

The website also consists of an administration area that allows staff to make changes to content that is publicly accessible. This project does not cover any optimization of the administration area. However, dependencies between public pages and admin pages will have to be taken into account when modifying scripts in the user area.

1.1.2. Use case diagram

The following figure illustrates the use case diagram of the website. It presents the logic as it is perceived by end users. The refactoring process will not affect the application logic on this level of abstraction, as refactoring is “any change to a computer program which improves its readability or simplifies its structure without changing its results” (Wikipedia Foundation, Inc. 2007).

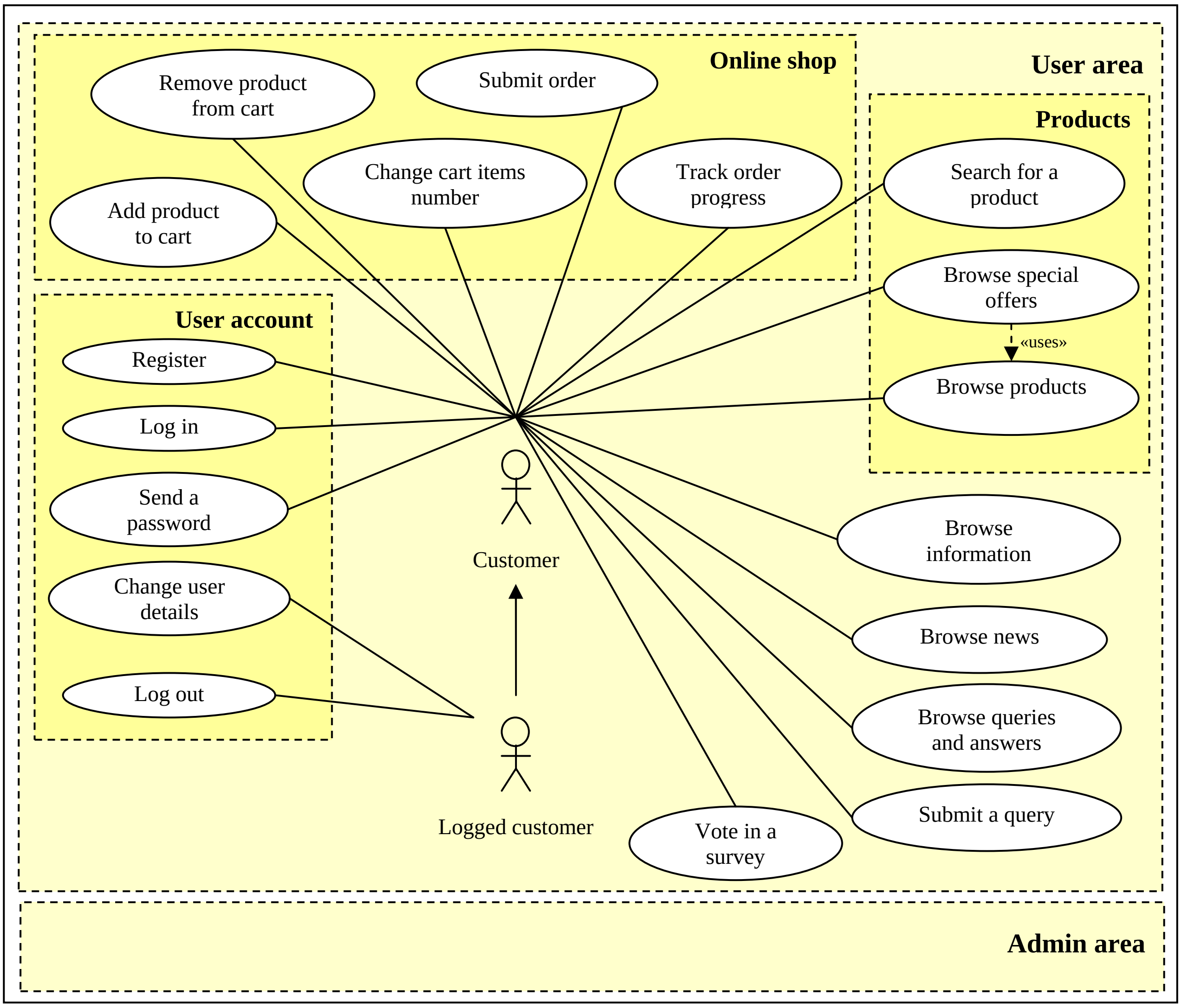


Figure 1.1.2. Use case diagram of the website

1.1.3. Physical structure – files

Figure 1.1.3 displays the structure of the website before optimization. Almost all pages in the user area are accessible through the index.php file. For example, to access the page with product category that has an ID number 250, the following link would be used:

```
/index.php?xx=2&hl=250&zo=&pod=250
```

The index.php page knows that if the `xx` parameter equals 2, it should include modules that display products. The `hl` and `pod` parameters tell the script the ID of the category. The `zo` parameter may be used to set the type of view (brief or full).

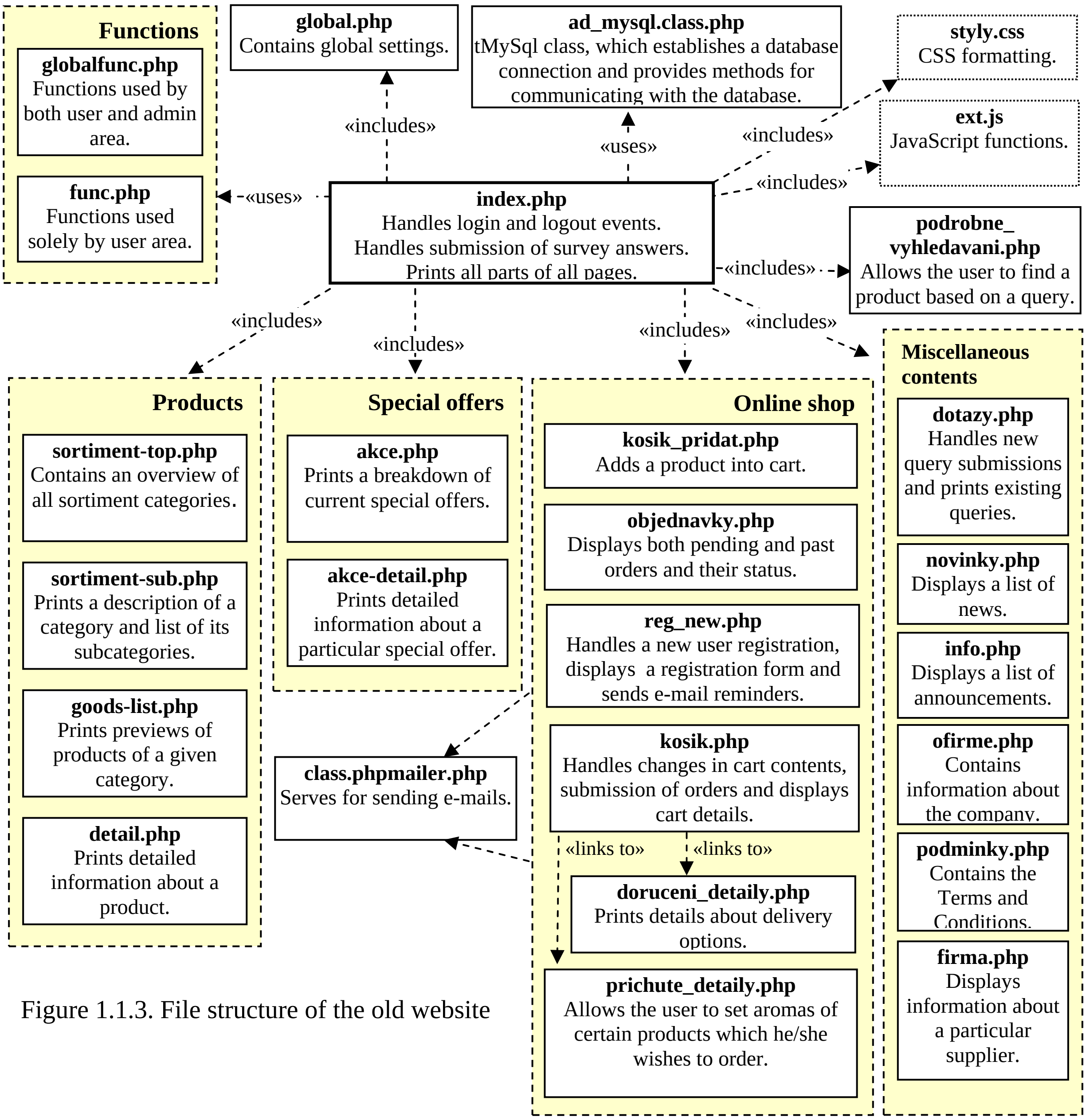

Functions
globalfunc.php
Functions used by both user and admin area.
func.php
Functions used solely by user area.
global.php
Contains global settings.
ad_mysql.class.php
tMySql class, which establishes a database connection and provides methods for communicating with the database.
styly.css
CSS formatting.
ext.js
JavaScript functions.
«includes»
«uses»
«includes»
«includes»
«uses»
index.php
Handles login and logout events. Handles submission of survey answers. Prints all parts of all pages.
«includes»
podrobne_ vyhledavani.php
Allows the user to find a product based on a query.
«includes»
«includes»
«includes»
«includes»
Products
sortiment-top.php
Contains an overview of all sortiment categories.
sortiment-sub.php
Prints a description of a category and list of its subcategories.
goods-list.php
Prints previews of products of a given category.
detail.php
Prints detailed information about a product.
Special offers
akce.php
Prints a breakdown of current special offers.
akce-detail.php
Prints detailed information about a particular special offer.
class.phpmailer.php
Serves for sending e-mails.
Online shop
kosik_pridat.php
Adds a product into cart.
objednavky.php
Displays both pending and past orders and their status.
reg_new.php
Handles a new user registration, displays a registration form and sends e-mail reminders.
kosik.php
Handles changes in cart contents, submission of orders and displays cart details.
«links to»
«links to»
doruceni_detaily.php
Prints details about delivery options.
prichute_detaily.php
Allows the user to set aromas of certain products which he/she wishes to order.
Miscellaneous contents
dotazy.php
Handles new query submissions and prints existing queries.
novinky.php
Displays a list of news.
info.php
Displays a list of announcements.
ofirme.php
Contains information about the company.
podminky.php
Contains the Terms and Conditions.
firma.php
Displays information about a particular supplier.

Figure 1.1.3. File structure of the old website

Relation	Description
«includes»	The module to which the arrow points is included within the parent module by means of either the `include()` or `require()` function. In both cases it is a simple pasting of code into the parent module. This code usually prints some text directly to the output.
«uses»	The module to which the arrow points is included as in the case of the «includes»

	relation. However, in this case the included module does not contain only a sequence of commands. The commands in this case are enveloped either in functions or methods (in case of objects) which must be first invoked to produce some result.
«links to»	The module to which the arrow points can be accessed via a link contained in the parent module (using either the standard <a> element or a javascript).

Please note that figure 1.1.3 does not show all relations to keep the diagram simple and clear.

1.2. Drawbacks of the old design

1.2.1. Security

First we look briefly at an excerpt from the old index.php file that handles the user request to log off:

```
25 if(trim($login)=="odhlas"){
26   $data_odhlasit = array("session"=>"");
27   $zmenit_reg = $sql->update("users", $data_odhlasit, "id=".$od_id."");
28 }
```

On line 25, the script expects the **`register_globals`** setting in the php.ini file to be enabled because the `$login` variable comes from the GET request. However, this option has several security issues and is implicitly disabled as of PHP 4.2.0 (The PHP Group 2007). Instead of `$login`, we should use the GET superglobal array: `$_GET['login']`.

Line 27 presents another security issue. The `$od_id` variable is obtained from a GET request and is passed to the `$sql` object without verifying that it does not include a **SQL injection**. Ideally, the `$sql` object would do the testing internally but this is not the case either.

1.2.2. Encapsulation

Another problem arises from the fact that the script directly accesses the database, even if by means of the `$sql` object. The drawback of this solution is that the script must know the names of the relevant tables and its fields. If these were to change, the script would have to be rewritten as well. In this simple example, this seems to cause no problem. However, if we consider that the table is queried on many different places and not only by the index.php, any changes to the structure of the table would be extremely difficult to reflect in the code that accesses it. The programmer would then be very likely to commit an error. It may be argued that changes to the structure of database should be rare, provided the initial database design was well thought through. Despite this, modifications may be necessary in practise. Clearly, any inconsistencies springing from a change in the database design can be minimized by encapsulating the access to the `users` table into a class that will represent the User entity.

1.2.3. MVC (Model-View-Controller)

From the MVC point of view, the index.php page contains the model, view and controller intermingled together. Actually, there is no concept of the MVC at all. The commands are in most cases contained directly in the page (inline scripting) or they are encapsulated in a function. The HTML output and the functions that access and modify data in the database are interwoven.

1.3. Implementing improvements

Most of the issues outlined above can be resolved by adopting the object oriented paradigm. From the MVC perspective, the classes and their methods constitute the Model. There will be both generic classes that simplify the most common and repetitive tasks (such as querying a database and processing the results) as well as classes that will represent a simplification of real-world entities, such as Customer, Product and Special offer. In the latter case, the class will provide a `load()` method that will fetch relevant data from a database and store them as attributes of an instance of the class. Similarly, to reflect any modifications subsequently done upon the attributes, the class will provide a `save()` method that will synchronize the variables of the given object with their relevant database counterparts.

1.3.1. Creating generic classes

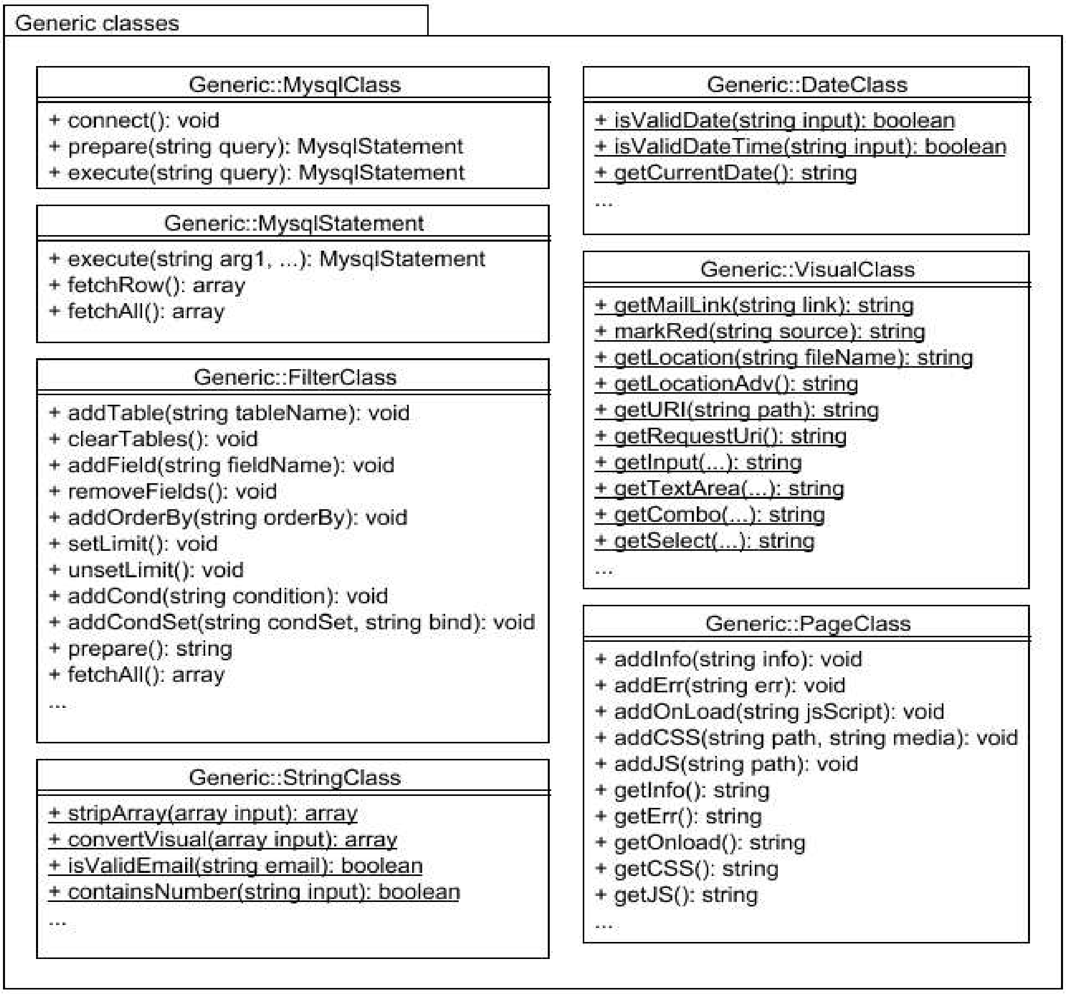

Generic classes
Generic::MysqlClass
+ connect(): void
+ prepare(string query): MysqlStatement
+ execute(string query): MysqlStatement
Generic::MysqlStatement
+ execute(string arg1, ...): MysqlStatement
+ fetchRow(): array
+ fetchAll(): array
Generic::FilterClass
+ addTable(string tableName): void
+ clearTables(): void
+ addField(string fieldName): void
+ removeFields(): void
+ addOrderBy(string orderBy): void
+ setLimit(): void
+ unsetLimit(): void
+ addCond(string condition): void
+ addCondSet(string condSet, string bind): void
+ prepare(): string
+ fetchAll(): array
...
Generic::StringClass
+ stripArray(array input): array
+ convertVisual(array input): array
+ isValidEmail(string email): boolean
+ containsNumber(string input): boolean
...
Generic::DateClass
+ isValidDate(string input): boolean
+ isValidDateTime(string input): boolean
+ getCurrentDate(): string
...
Generic::VisualClass
+ getMailLink(string link): string
+ markRed(string source): string
+ getLocation(string fileName): string
+ getLocationAdv(): string
+ getURI(string path): string
+ getRequestUri(): string
+ getInput(...): string
+ getTextArea(...): string
+ getCombo(...): string
+ getSelect(...): string
...
Generic::PageClass
+ addInfo(string info): void
+ addErr(string err): void
+ addOnLoad(string jsScript): void
+ addCSS(string path, string media): void
+ addJS(string path): void
+ getInfo(): string
+ getErr(): string
+ getOnload(): string
+ getCSS(): string
+ getJS(): string
...

Figure 1.3.2. Class diagram for generic classes

MysqlClass: This class provides an interface for accessing a database. It ensures that a possible SQL injection will be dealt with accordingly. This class is used in combination with the MysqlStatement class.

VisualClass: Can be used to create HTML output more effectively.

PageClass: It allows the invoker to set various page properties.

1.3.2. Creating shop and information classes

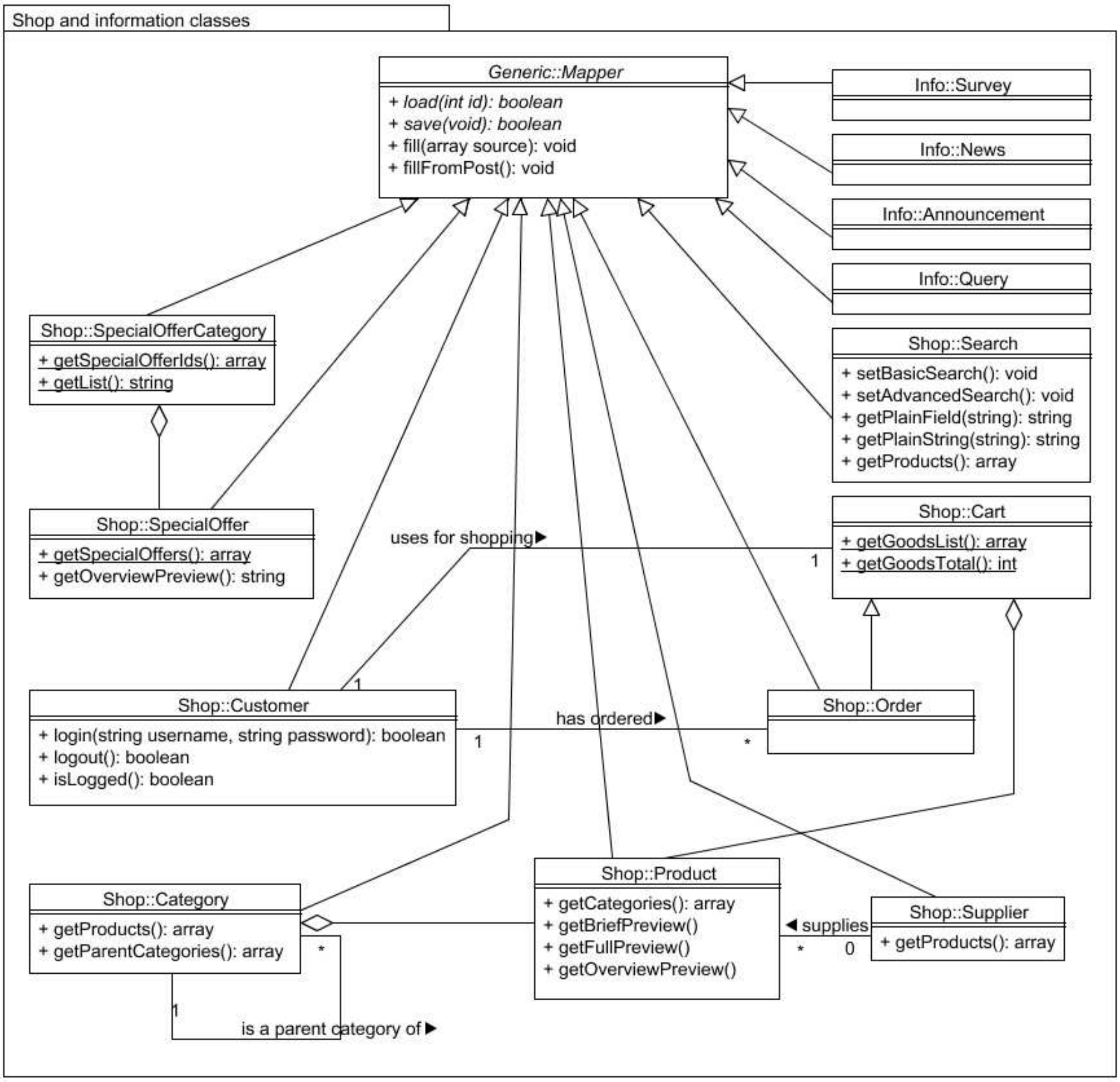


Figure 1.3.1. Class diagram for shop and information classes

On figure 1.3.1, the class diagram is shown. There are several aspects to clarify: The PHP does not support multiple inheritance. The Mapper class, in fact, contains many other methods that have identical definitions for SpecialOffers and Products. Ideally, there would be one Mapper parent class and another Goods parent class. The Products and SpecialOffers would be a specialization of both

Mapper and Goods. However, this is not possible in PHP and therefore there is only one Mapper parent class that encompasses all methods that have the same definition in at least two child classes.

There are also methods that have not been implemented. For example the combination of `fillFromPost()` and `save()` methods would be utilized in the admin area to update the relevant record in the database.

Also, the News, Information and Query classes have not been re-implemented. Considering the extent of the application and the fact that the main goal of this project was to optimize the front-end, it was necessary to choose trade-offs and refactor only those scripts whose optimization was likely to speed up the process of optimizing the front-end output. The scripts that handle news, information and customer queries are fairly isolated and easy to modify even without refactoring.

On the other hand, the scripts that manipulate products, special offers, categories and suppliers appear to be the best candidates for refactoring. These scripts make up the core of the online shop and presumably, these will require extensive SEO optimization in later stages of the project. The author therefore focused on refactoring of the following classes: Product, Category, Customer, Supplier, SpecialOffer, SpecialOfferCategory and Search.

1.3.3. Case study: The logout procedure

This section will demonstrate how the insecure logout procedure from section 1.2.1 has been transformed and how it relates to encapsulation. First look at an extract from the Customer class:

```
48 public function logout($PHPSESSID){
49   $sqlUpdate  = "UPDATE users SET session='' ";
50   $sqlUpdate .= "WHERE (session!='' AND session IS NOT NULL AND
                    session=':01')";
51
52   self::$dbh->prepare($sqlUpdate)->execute($PHPSESSID);
53 }
```

The `logout()` method expects a session identifier on input and then it updates the corresponding record in the database, causing the user to be marked as logged out. The important thing is that the SQL statement on line 50 only contains the ":01" string instead of the actual value.

Looking at line 52, the `self::$dbh->prepare($sqlUpdate)` command returns an instance of the `MysqlStatement` class. Invoking the `execute($PHPSESSID)` method upon this instance results in replacing the ":01" string by the actual value of the `$PHPSESSID` variable. The `MysqlStatement` ensures that the `$PHPSESSID` variable will be tested for possible SQL injection. This approach of pre-processing the SQL statement first and dealing with potentially insecure parameters afterwards has two advantages:

a) The invoker needs not to test dangerous inputs; it can delegate this work to the `execute()` method.

b) The invoker may prepare a template SQL statement and then execute it repeatedly with different parameters.

The `logout()` method is encapsulated in the Customer class and invoking this method is the only way for a customer to log off. If the structure of the `users` table was to change, the programmer would only need to change this method.

The idea of pre-processing MySQL statements has been borrowed from Schlossnagle (2004).

1.3.4. The MVC - theory

The MVC (Model-View-Controller) is an architectural pattern that simplifies the maintenance of large software applications. The basic idea is to split the application into several layers and define their interfaces so that changes in internal structure of one layer will not require modifying the internal implementation of another layer, as the interfaces remain the same.

In web applications, the Model represents the engine that manipulates the application data, e.g. data in a database. The view constitutes the front-end, in other words how the information obtained from the Model is presented to the end user. Finally, there is the Controller. This entity responds to user requests, as a result of which it may invoke the Model's methods. The View exploits the Model to generate its output but the Model does not know about the View.

The following diagram depicts the MVC schematically. The solid lines indicate a direct association, and the dashed lines indicate an indirect association (Wikipedia Foundation, Inc. 2006a).

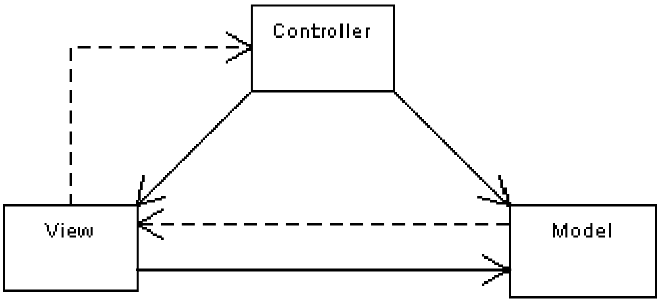


Figure 1.3.4 Model-View-Controller (Wikipedia Foundation, Inc. 2006b)

1.3.5. The MVC in the HC Compact website explained on example

The Model-View-Controller concept in the optimized HC Compact website will be explored by means of the following page:

```
/index.php?xx=2&pod=250
```

There are two reasons for choosing this page:

a) It is a page consisting of a listing of products of a category. This is the part of the website that has been completely refactored and as such it is designed according to the MVC pattern.

b) It is a page where the most complicated dependencies can be explained.

In what follows we drill down into the logic flow, starting from the very URL. The index.php file can be labelled as a front-end controller. For the user, it is the access point to the website. Looking at line 6 of the index.php file, we can see that it makes use of the globalinit.php file:

```
6 require_once($pagePrefix."classes/globalinit.php");
```

The globalinit.php file is the controller. Looking into the globalinit.php file, we can see that it uses two types of scripts: It includes script from the `/include` directory and from the `/classes` directory. The PHP scripts from the `/include` directory contain parts of the decomposed controller. The PHP scripts from the `/classes` directory contain the model. So, the globalinit.php (the controller) accesses the model, which is one of the ideas of the MVC.

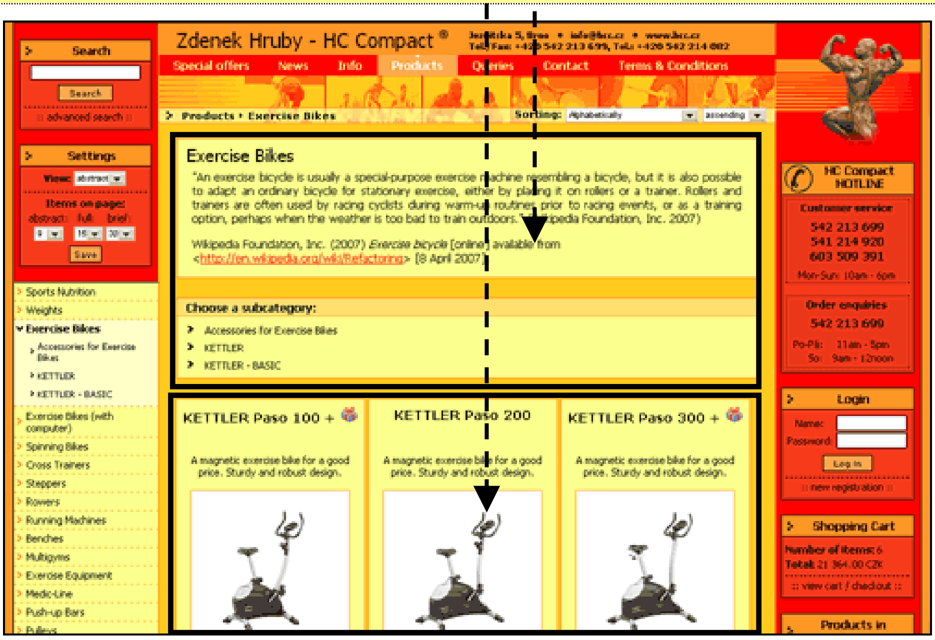
Going back to the index.php file, we can see that it includes the view part of the MVC, on line 9:

```
9 include($pagePrefix."layout/layout.php");
```

Going into the layout.php file, we can see that this module is made up of a layout structure that is common for all pages in the user area (except of popup windows). For the sake of simplicity, the module is further decomposed into several smaller parts that the layout.php module includes. Taking the example of the above stated URL, where the `xx` parameter equals 2, the layout.php module includes two groups of templates: templates from the `/layout` directory and from the `/sortiment` directory. The `/layout` directory contains templates that are the same for all pages (left-hand column, right-hand column and the title strip), whereas the `/sortiment` directory contains templates that are specific for all pages with product listings.

Now we will examine the modules that reside in the `/sortiment` category in more detail. The following figure depicts the modules included in the layout.php file and their output:

```
18 include("sortiment/goods-list-engine.php");
19 include("sortiment/sortiment-sub.php");
20 include("sortiment/goods-list.php");
```



Exercise Bikes

"An exercise bicycle is usually a special-purpose exercise machine resembling a bicycle, but it is also possible to adapt an ordinary bicycle for stationary exercise, either by placing it on rollers or a trainer. Rollers and trainers are often used by racing cyclists during warm-up routines prior to racing events, or as a training option, perhaps when the weather is too bad to train outdoors." (Wikipedia Foundation, Inc. 2007)

Wikipedia Foundation, Inc. (2007) *Exercise bicycle* [online] available from <http://en.wikipedia.org/wiki/Refactoring> [8 April 2007]

Choose a subcategory:

- Accessories for Exercise Bikes
- KETTLER
- KETTLER - BASIC

KETTLER Paso 100 +

A magnetic exercise bike for a good price. Sturdy and robust design.

KETTLER Paso 200

A magnetic exercise bike for a good price. Sturdy and robust design.

KETTLER Paso 300 +

A magnetic exercise bike for a good price. Sturdy and robust design.

Figure 1.3.5 View modules explained

Note that the goods-list-engine.php file contains the view logic and is therefore not directly visible in the output. What it basically does is that it calls some class in order to obtain an array of instances of products. These are then used by the goods-list.php which iterates through these instances and prints a box for each product on the output. The idea here is that the view part of the MVC is even further split into the view logic and the view layout. Another thing to point out is that all of these three modules access some classes to get data from them but never invoke those class methods that would change the model, e.g. update some data in the database. Such modifications can be conducted only by the controller.

1.3.6. The MVC in the HC Compact website in more abstract terms

Now that the MVC has been demonstrated on an example, we can think of the MVC in the HC Compact site in relation to the file structure and dependencies among the files.

In figure 1.3.6, the arrows display the workflow. It can be seen that first the index.php is called, which passes control to the controller that handles the request, often by changing the model (invoking class methods). The model can internally access the database, hence the fourth step. Then, the view follows, which obtains data from the model and formats them. This output is finally presented to the user through the front controller (the index.php file).

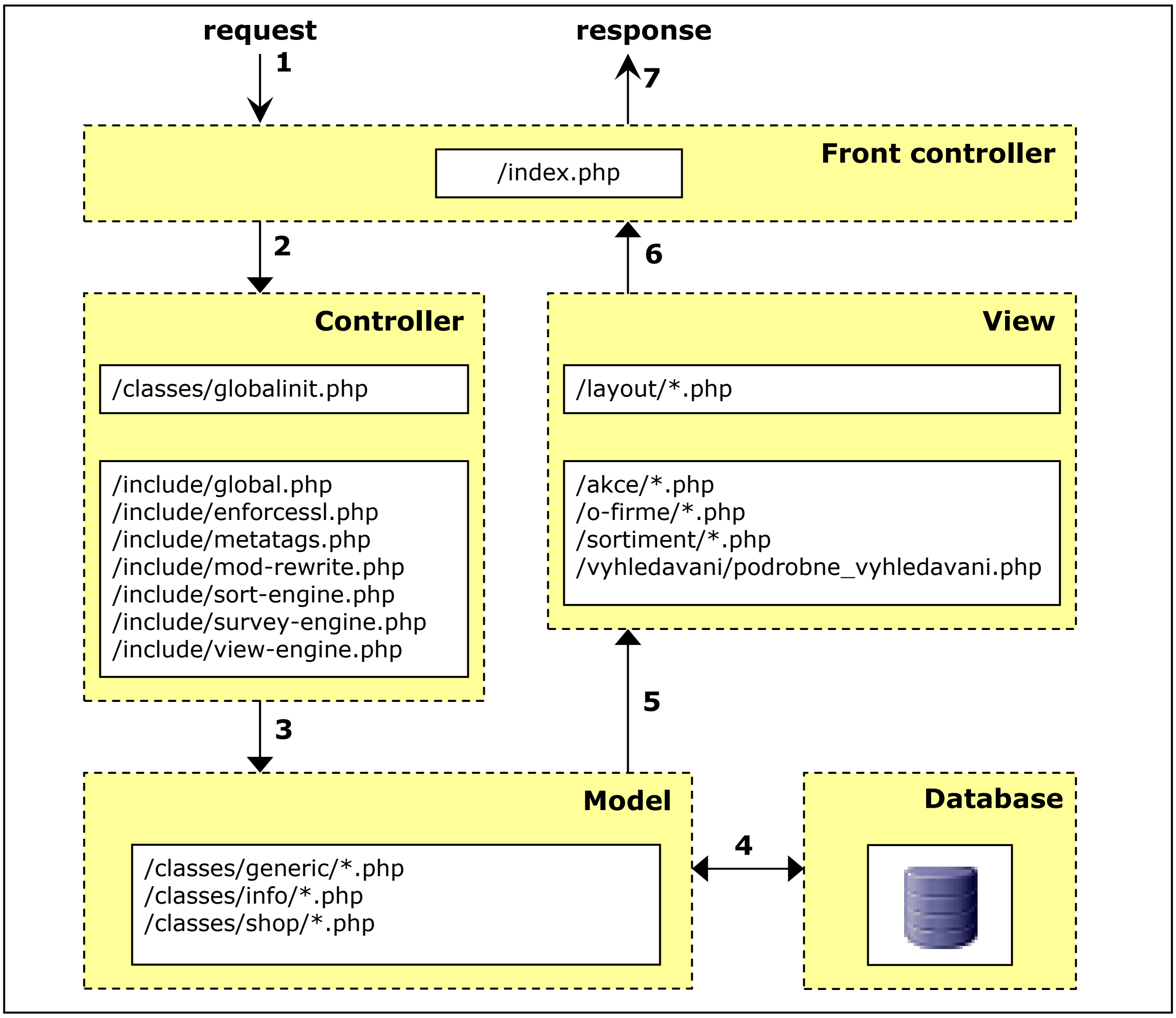


Figure 1.3.6. MVC in the HC Compact website

Note that some PHP files of the website are not displayed in the diagram. The reason for this is that they have not been refactored to reflect the MVC principles. Also, this scheme does not include CSS and Javascript files. In fact, they form a part of the view but for the sake of descriptiveness they are left out from this diagram.

1.4. Summary

In the first part of the project, the major flaws in the design of the back-end of the existing application have been identified. It can be argued that the old design was sufficient in the early stages of the project because at that time, the application was not so extensive.

Nevertheless, the application today is a large-scale one and needed refactoring. The improvements that are to be undertaken have been demonstrated and partially implemented, in particular where the odds were that it accelerates further work on this project. However, there still remain sections written purely in the procedural paradigm. It is the judgement of the author that a complete refactoring of the entire user area is out of the scope of this project.

2. Front-end: Compliance to W3C standards

2.1. Web Content Accessibility Guidelines 1.0

The project set the target for the website to conform to the Web Content Accessibility Guidelines (WCAG) 1.0. These guidelines can be accessed at http://www.w3.org/TR/WAI-WEBCONTENT/. In what follows, the differences between accessibility and usability will first be explained, putting them into relation with the above stated document.

2.1.1. Usability versus accessibility

The basic difference between these two words can be derived from their very meaning: if a page is accessible, people are able to access and use its content. Primarily, accessibility focuses on people with disabilities (Thatcher 2002). A page being accessible for a sight-impaired person using a voice browser means that the person can access the content. However, accessible pages are often of benefit to people without disabilities as well. A typical example may be an alternative text (`alt` attribute) of images (`img` elements). Supplying the alternative text will be both beneficial for a blind person using a voice browser, as well as for a sighted person using a text browser, such as Lynx.

Usability can be described as an “added value” to accessibility. If a website is designed according to the ideas of usability, users are likely to find such a website satisfying, because they can work with it efficiently and learn its logic very quickly. As far as people with disabilities are concerned, these are affected by a website with poor usability to the same extent as people without disabilities.

2.1.2. WCAG 1.0 conformance levels

The Web Content Accessibility Guidelines (WCAG) state the requirements that a website must or should follow in order to comply with the WCAG. The requirements are broken up into three levels with different priorities. The accessibility issues have the highest priority, whilst usability issues are of lower priority. An exact definition of priorities and their fulfilment can be found at http://www.w3.org/TR/WCAG10/full-checklist.html.

The following sections will systematically cover all WCAG priority 1 and 2 requirements and describe the improvements implemented in the HC Compact website.

2.1.3. Priority 1 checkpoints

Checkpoint 1.1: Provide a text equivalent for every non-text element

The HC Compact website contains only images as a non-textual means of conveying information. The new website satisfies this guideline in that it provides an `alt` attribute for all `img` elements.

It should be pointed out that the website contains plenty of images defined in an external CSS files, using the `background-image` attribute. This applies e.g. for list bullets or images that form part of the layout. Obviously, there is no means to provide an alt attribute for these images. However, this is not needed, as images defined in a CSS file should inherently form part of a design. They should not carry any factual information and therefore there is no need for them to have a textual equivalent. The only issue to decide here is whether an image under consideration is part of the semantic contents of the website or part of the website’s design. Figure 2.1.3a illustrates the differences.

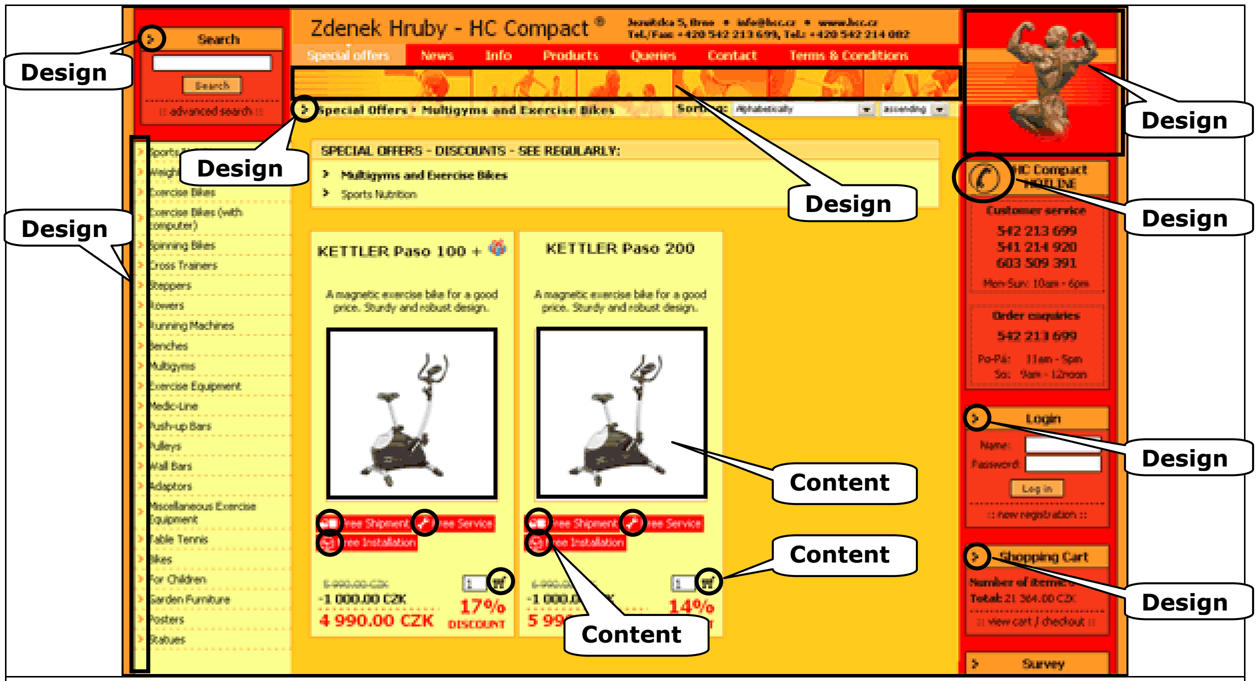


Figure 2.1.3a Images that convey information vs. images that form the design

Please note that the differences between content and its presentation are sometimes next to none. The picture proposes one possible solution but does not claim to be the only possible one.

Checkpoint 2.1: Ensure that all information conveyed with colour is also available without colour

The old website did not adhere to this rule, as it contained a registration form and shopping basket where required fields were distinguished from non-required fields solely by means of using red colour. This has been fixed by supplying an asterisk to each required field.

Checkpoint 4.1: Clearly identify changes in the natural language of a document's text and any text equivalents.

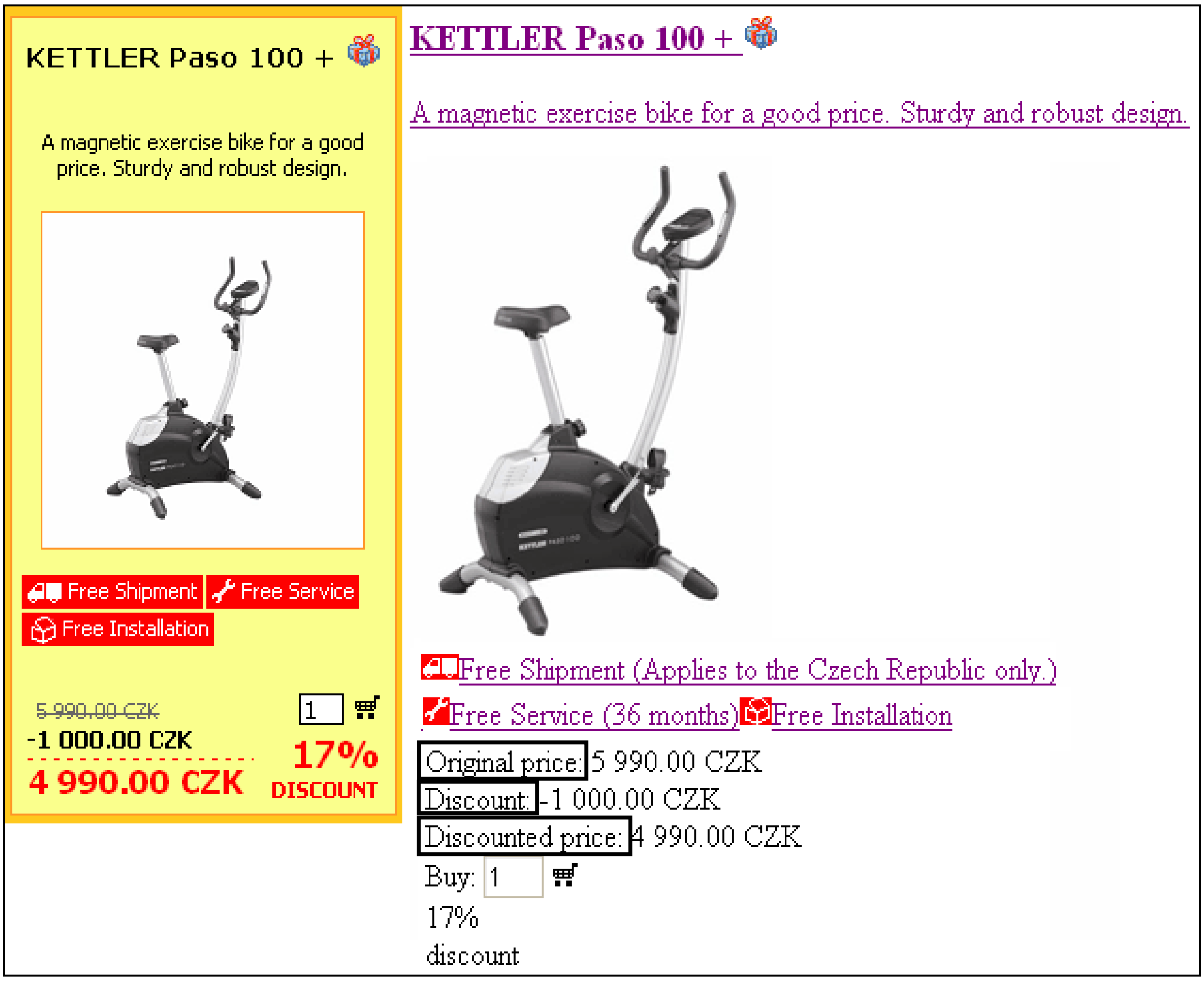
There are no bilingual sections on the website.

Checkpoint 6.1: Organize documents so they may be read without style sheets.

There are several aspects to point out considering the layout of the document when CSS are disabled. Firstly, there are short text descriptions throughout the website that are hidden when CSS are turned on. This applies for example for the “original price”, “discount” and the “discounted price” texts. When CSS are applied, the original price is crossed out, then the discount follows, and finally the discounted price is shown as a result of a subtraction under a line. When CSS are disabled, all three numbers appear as a plain text. Therefore, the document contains additional hints before the actual number to make it easier for the user to understand the meaning. Figure 2.1.3b demonstrates the differences.

Also, the left-hand navigation menu can be easily accessed when CSS are disabled, as it consists of a two-level unordered list (UL) of links. The old website's left-hand menu, on the other hand, did not clearly differentiate the first and second level of items, which may have been confusing for users with voice or text browsers.

When CSS are disabled, the website also provides two links to make it quicker to navigate on the page – "skip navigation" and "skip main content". This allows users with voice browsers to quickly get to the desired part of the page.



KETTLER Paso 100 +

A magnetic exercise bike for a good price. Sturdy and robust design.

Free Shipment Free Service

Free Installation

~~5 990.00 CZK~~

-1 000.00 CZK

4 990.00 CZK

17% DISCOUNT

KETTLER Paso 100 +

A magnetic exercise bike for a good price. Sturdy and robust design.

Free Shipment (Applies to the Czech Republic only.)

Free Service (36 months) Free Installation

Original price: 5 990.00 CZK

Discount: -1 000.00 CZK

Discounted price: 4 990.00 CZK

Buy: 1

17%

discount

Figure 2.1.3b Displaying the content with and without CSS formatting

Checkpoint 6.2: Ensure that equivalents for dynamic content are updated when the dynamic content changes.

The HC Compact website does not contain frames or applets. In regard to Java Scripts which generate dynamic contents, such as explanatory bubbles that appear over icons with gifts when hovered on, these texts are duplicated in the `alt` attributes of the corresponding icon that depicts the gift.

Checkpoint 7.1: Until user agents allow users to control flickering, avoid causing the screen to flicker.

There is no page that would flicker on the HC Compact website.

Checkpoint 14.1: Use the clearest and simplest language appropriate for a site's content.

The website now uses five levels of headings (H1…H5) to make it simpler for the user to skim the text and find information quickly if CSS are disabled. Next, all links contain a sensible anchor text that identifies the target. All links whose anchor text was just "here" have been altered in order to allow users to jump from link to link without reading the surrounding text (which is a common provision of voice browsers).

The WCAG also requires the following: limiting each paragraph to one main idea, avoiding slang, jargon, using active rather than passive verbs and avoiding complex sentences.

These requirements are unfortunately not very well quantifiable, as the Gunning fog index cannot be used to analyze Czech writing. Moreover, the author has not the right to amend all texts on the website, in particular the content of news, information, customer queries, company information and terms and conditions. It is the job of other employees of HC Compact to satisfy this requirement. As for the English version of the website developed for purposes of this project, it does fully satisfy this checkpoint.

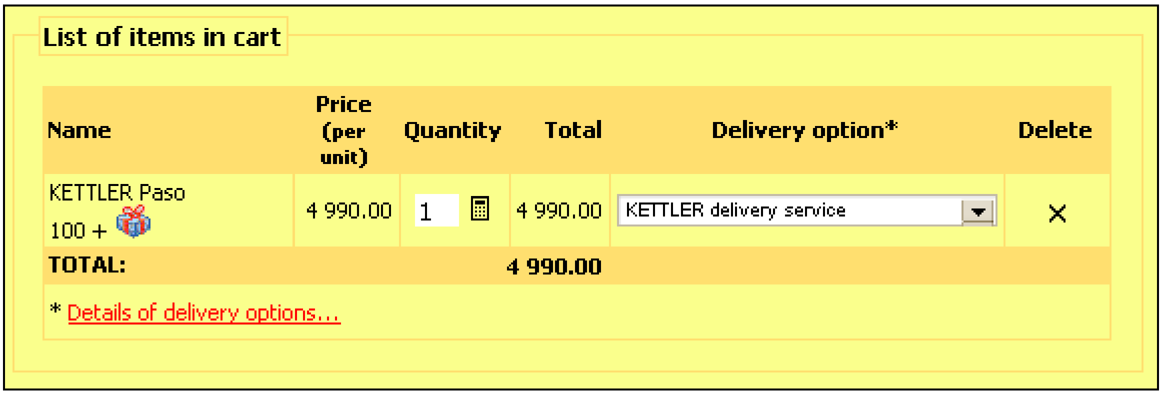
Checkpoint 5.1: For data tables, identify row and column headers.

Checkpoint 5.2: For data tables that have two or more logical levels of row or column headers, use mark-up to associate data cells and header cells.

These points require that a table makes it clear for a voice browser where to find headers for data columns. There are three attributes that can be used to help assistive technologies to make this out: `scope`, `headers` and `axis`. The first one can be used to denote whether a TH element refers to a row of data cells or a column of data cells. The `headers` and `axis` attributes come useful with complex tables that convey information consisting of more than two dimensions.

These two checkpoints also require structural groups of rows to be grouped using the THEAD, TFOOT and TBODY elements, and groups of columns to be grouped using the COLGROUP and COL elements.

In what follows, it will be demonstrated how this point has been satisfied in the case of a table that displays a list of goods contained in the shopping basket. First examine a screenshot and the corresponding HTML code:



List of items in cart

Name	Price (per unit)	Quantity	Total	Delivery option*	Delete
KETTLER Paso 100 +	4 990.00	1	4 990.00	KETTLER delivery service	×
TOTAL:			4 990.00		

* Details of delivery options…

Figure 2.1.3c Identifying rows and columns in a table

```
<table>
<colgroup>
  <col width='23%'><col width='12%'><col width='15%'><col width='12%'>
  <col width='10%'><col width='13%'><col width='15%'>
</colgroup>

<thead>
  <tr>
    <th scope='col'>Name</th>
    <th scope='col'>Price <span class='small'>(per unit)</span></th>
    <th scope='col'>Quantity</th>
    <th scope='col'>Total</th>
    <th scope='col'>Delivery option*</th>
    <th scope='col'>Delete</th>
  </tr>
</thead>

<tfoot>
  <tr>
    <th scope='row'>TOTAL:</th>
    <td colspan='4'>4 990.00</td>
    <td colspan='2'></td>
  </tr>
</tfoot>

<tbody>
  <tr>
    <td>KETTLER Paso 100...</td>
    <td>4 990.00</td>
    <td><input ...><input ...></td>
    <td>4 990.00</td>
    <td><select>...</select></td>
    <td><input ...></td>
  </tr>
</tbody>

</table>
```

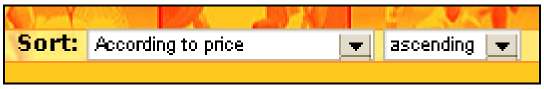
The above code demonstrates how the `scope` attribute should be used in order to convey the right direction for linearizing. Also, corresponding rows are grouped together, using the `thead`, `tbody` and `tfoot` elements.

Checkpoint 6.3: Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported.

In order to satisfy this requirement, several changes had to be done. The following examples demonstrate two issues which had to be addressed:

Forms that automatically submit themselves

The website made use of several pull-down menus that were automatically submitted when the selected option changed. The user did not have to click on a submission button. In fact, he *could* not, as there was no submission button whatsoever.



Sort: According to price ascending

Figure 2.1.3d Pull-down menus that automatically submit themselves.

Looking at the code for the first pull-down menu before the optimization, we would find this:

```
<select onchange='window.location="/include/sort-
engine.php?sort="+this.value+"&returnURI=%2F"'>
```

What is to point out here is that the `select` element has no name and is not enclosed in any `form` element. The submission works but only if java scripts are enabled.

The code has been optimized as follows (now both pull-down menus from the screenshot 2.1.3d are included):

```
<form method='get' action='/include/sort-engine.php'>

<select name='sort' onchange='window.location="/include/sort-
engine.php?sort="+this.value+"&returnURI=%2Fhcc%2F"'>...</select>

<select name='sortHow' onchange='window.location="/hcc/include/sort-
engine.php?sortHow="+this.value+"&returnURI=%2Fhcc%2F"'>...</select>

<input type='hidden' name='returnURI' value='/'>

<span class='hideByJS'>
  <input type='submit' value='OK' class='button'>
</span>

</form>
```

Note that what has been added appears in bold.

Now the form can be submitted regardless of whether java scripts are enabled or disabled. There is only one, rather minor problem to deal with: The submission button should not be visible if java scripts are enabled because all submissions are done automatically and it would be of no use. To do this, a `span` element with class attribute set to `hideByJS` encloses the submission button. Looking into the `/include/interaction.js` module, we find out that the class name is used as an indicator for the java script to hide the element:

```
if(inputs[i].className.indexOf("hideByJS")!=-1){
  inputs[i].className += " hidden";
}
```

This excerpt forms a part of the `addListeners()` function that is invoked immediately after the page has been loaded.

It can be seen that the java script only sets another CSS class to the element. Finally, we have to look into the `/include/globalstyles.css` module:

```
.hidden {display: none;}
```

Using the approach described will thus hide the redundant submission button only if JavaScript is enabled.

Popup windows

Popup windows are windows that open up as dialog boxes using the `window.open` java script function. The user must be presented an equivalent functionality if scripting is suppressed. The following snippet shows how to do that:

```
<a href="link" onclick="return !openWindow('link', width, height);">anchor text</a>
```

The `openWindow` function internally exploits the `window.open` function as follows (code from the `/include/ext.js` file):

```
newWindow = window.open(...);

return newWindow!=null;
```

The result is that if scripting is enabled, a popup window is opened up, causing the `onclick` inline script to return false, as a result of which the ordinary link (specified by the `href` attribute) is ignored. On the other hand, if scripting is disabled, the `onclick` inline script returns true and the ordinary link will be opened up as a regular page.

2.1.4. Priority 2 checkpoints

Checkpoint 2.2: Ensure that foreground and background colour combinations provide sufficient contrast when viewed by someone having colour deficits or when viewed on a black and white screen.

To determine whether the contrast is sufficient, the colour space of several page screenshots was reduced to greyscale and the contrast appears to be sufficient when scrutinized. This point would ideally require a user testing with sight impaired people but this would overlap the extent of this project.

Checkpoint 3.1: When an appropriate mark-up language exists, use mark up rather than images to convey information.

The website does not contain images representing text. Also, formatting and layout is done purely by using CSS, as the WAI recommends in the thorough description of this checkpoint.

Checkpoint 3.2: Create documents that validate to published formal grammars.

Looking at the first line of each HTML page, we can find out that the website declares to be HTML 4.01 Strict valid:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

A testing has been accomplished to prove this, using the W3C online validation tool located at http://validator.w3.org/. The pages have been found valid.

Note that the initial project's specification stated that the website would adhere to XHTML 1.0 after optimization. This has been altered due to extensive use if java scripts that exploit the DOM (Document Object Model). If the XHTML was to be used, these scripts would have to be rewritten and tested, which would presumably cause plenty of compatibility problems (Langridge 2005).

Secondly, all external CSS files have been tested (using an online validation tool located at http://jigsaw.w3.org/css-validator/) and found valid. This applies for all CSS files that reside in the `/include/` directory.

There is, however, one CSS attribute that is not valid and has been used. Looking at the source code of any HTML page of the HC Compact website, it can be found that there are CSS definitions directly in a style element, starting with this line:

```
body {behavior: url('/hcc/include/csshover.htc');}
```

The `behaviour` element is a proprietary element used solely by MSIE and as such should be avoided, as other browsers do not support it. The `csshover.htc` script is a third-party script that allows a developer to use the `:hover` pseudo-class for LI elements. This behaviour should be commonly catered for in modern browsers, though MSIE 6.0 does not allow exploiting the `li:hover` statement. The author needed to use this pseudo-element for the left-hand menu and its hovering effects, hence decided to cope with this insufficient provision in MSIE by breaking the W3C standards.

The author is, however, convinced that using a proprietary element in this case does not hinder him from declaring this checkpoint as satisfied. The `behaviour` element is used only as a supplement for MSIE, as a secondary means in case the browser does not allow for a proper CSS definition with cross-browser support. A completely different case would be if a proprietary definition would be the only means to achieve some functionality, which would plainly be a step towards breaking W3C standards and its effort to make World Wide Web a cross-browser, platform-independent medium.

Checkpoint 3.3: Use style sheets to control layout and presentation.

This checkpoint basically requires to rigorously detach structure of a document from its presentation. Such documents allow better accessibility, manageability, and portability (W3C 2000). The „Core

Techniques for Web Content Accessibility Guidelines 1.0“ (W3C 2000) describe several techniques that are to follow:

- Sections of text should be identified with heading elements (H1-H6).
- Structural elements should not be used for presentational effects (such as usage of BLOCKQUOTE to achieve indentation).
- EM and STRONG elements should be used instead of B and I elements, as the latter ones were designed to create visual presentation effects, whereas EM and STRONG indicate structural emphasis that may be rendered in a variety of ways (font style changes, speech inflection changes).
- Layout, positioning, layering, and alignment should be done by means of style sheets.

(W3C 2000)

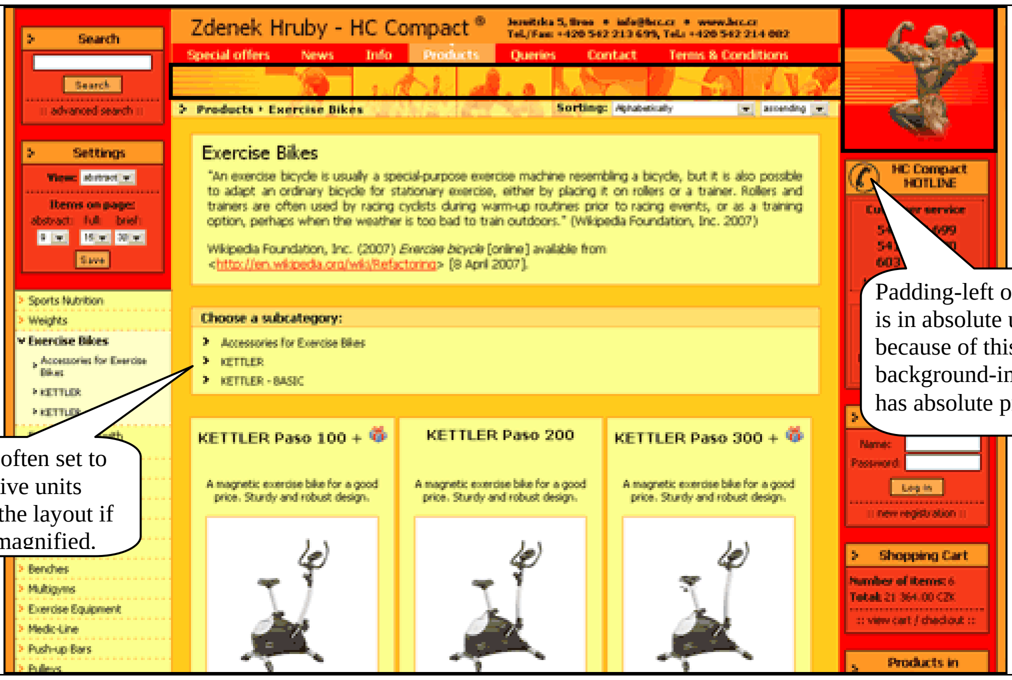
All the above stated requirements have been abided by when refining the front-end output. Rendering the document without CSS effects will have no impact on understandability of the website’s content.

Checkpoint 3.4: Use relative rather than absolute units in mark-up language attribute values and style sheet property values.

Users, and those with sight problems in particular, should be able to easily magnify the website’s font size, which will allow them to read all text without difficulties. From the developer’s point of view, this can be achieved by using relative units (em or percentage) rather than absolute units (px, pt, cm, etc.) in CSS definitions. In section „3 Units of measure“, the W3C (2000) also defines when it is still possible to use absolute units: „Only use absolute length units when the physical characteristics of the output medium are known, such as bitmap images.“

The HC Compact website after optimization still contains plenty of absolute units. However, it is the opinion of the author that these are well-founded, since the very layout is based on several bitmap images with fixed proportions, as illustrated in figure 2.1.4a. There are two images that dictate the width of the middle and right-hand column. To keep the layout balanced, the left-hand column has the same width as the right-hand column.

There are several other examples where absolute units had to be exploited, as the bubbles in figure 2.1.4a explain. It may be argued that the website does not conform to this checkpoint because of borders that are commonly defined with pixel units. However, it was found that horizontal and vertical lines are not rendered with the same thickness when magnified if the border thickness is specified by means of relative units. Therefore, this minor deviation does not constitute a sound reason for not declaring the website compliant with this checkpoint. The most important aspect, which is the provision of changing font size, has been fully dealt with in the new design.



Exercise Bikes

"An exercise bicycle is usually a special-purpose exercise machine resembling a bicycle, but it is also possible to adapt an ordinary bicycle for stationary exercise, either by placing it on rollers or a trainer. Rollers and trainers are often used by racing cyclists during warm-up routines prior to racing events, or as a training option, perhaps when the weather is too bad to train outdoors." (Wikipedia Foundation, Inc. 2007)

Wikipedia Foundation, Inc. (2007) *Exercise bicycle* [online] available from <http://en.wikipedia.org/wiki/Refactoring> [8 April 2007].

Choose a subcategory:

- Accessories for Exercise Bikes
- KETTLER
- KETTLER - BASIC

KETTLER Paso 100 +

A magnetic exercise bike for a good price. Sturdy and robust design.

KETTLER Paso 200

A magnetic exercise bike for a good price. Sturdy and robust design.

KETTLER Paso 300 +

A magnetic exercise bike for a good price. Sturdy and robust design.

Borders are often set to 1px, as relative units might spoil the layout if the view is magnified.

Padding-left of this box is in absolute units because of this background-image that has absolute proportions.

Figure 2.1.4a Fixed bitmap images

Checkpoint 3.5: Use header elements to convey document structure and use them according to specification.

Header elements are used as required by WCAG.

Checkpoint 3.6: Mark up lists and list items properly.

List items have been used in the appropriate way in the new design. A typical example is the left-hand navigation menu.

Checkpoint 3.7: Mark up quotations. Do not use quotation mark-up for formatting effects such as indentation.

There are no quotations used on the website.

Checkpoint 6.5: Ensure that dynamic content is accessible or provide an alternative presentation or page.

The website does not make use of frames, nor does it contain java scripts that would prevent the user from an action if these were disabled.

Checkpoint 7.2: Until user agents allow users to control blinking, avoid causing content to blink

There are no elements that would blink.

Checkpoint 7.4: Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.

The website does no contain any periodically auto-refreshing pages.

Checkpoint 7.5: Until user agents provide the ability to stop auto-redirect, do not use mark-up to redirect pages automatically. Instead, configure the server to perform redirects.

The old website made use of meta refresh in the course of adding a product into the shopping cart. This behaviour has been altered so that no automatic redirect is now used.

Checkpoint 10.1: Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user.

Some pages do exploit popup windows, though not as the only means to arrive at the given URL (this has been explained in checkpoint 6.3.). In regard to automatic changes of windows, this requirement has been satisfied in that the popup windows for choosing presents or aromas for a product placed in the shopping basket now contain a note informing the user about the refresh that is to take place when he or she closes the popup window.

Checkpoint 11.1: Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported.

Currently (July 2007), the latest version of HTML is the 4.01 Strict version and this has been used. As for styling, CSS 2.0 has been used, as this is the latest version widely supported by today’s browsers.

Checkpoint 11.2: Avoid deprecated features of W3C technologies.

Elements that are deprecated in HTML 4.01 are the following: APPLET, BASEFONT, CENTER, DIR, FONT, ISINDEX, MENU, S, STRIKE, U. The HC Compact website does not contain any of them after optimization.

Checkpoint 12.3: Divide large blocks of information into more manageable groups where natural and appropriate.

This requirement has been satisfied even in the old design. Examples of this can be seen on the shopping basket page, where FIELDSET and LEGEND elements are used to group similar items together.

Checkpoint 13.1: Clearly identify the target of each link.

Anchor texts have been revised and do not consist solely of ambiguous phrases like „click here” which are misleading when read out of context.

Checkpoint 13.2: Provide metadata to add semantic information to pages and sites.

The META elements and TITLE element of all pages have been refined to contain specific information about the particular page. META and TITLE elements are set in the `/include/metatags.php` module.

Checkpoint 13.3: Provide information about the general layout of a site (e.g., a site map or table of contents).

A sitemap has been created to meet this point. It is located at `/vyhledavani/mapa-stranek.php`

Checkpoint 13.4: Use navigation mechanisms in a consistent manner.

The website accommodates a consistent navigation that is the same across all pages.

Checkpoint 5.3: Do not use tables for layout unless the table makes sense when linearized.

The old website used tables for layout. This has been revised in the new version and now only CSS in combination with DIVs are used to lay out the site's elements.

Checkpoint 10.2: Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned.

Checkpoint 12.4: Associate labels explicitly with their controls.

All `label` elements have been explicitly associated with their input element if the label did not precede the element, in which case browsers should be able to infer the association implicitly. An example of an explicit association is given below:

```
<input id="sledovatZmeny" type="checkbox" name="sledovatZmeny" value="1">
<label for='sledovatZmeny'>
  I wish to be informed about the progress of the order by e-mail.
</label>
```

Checkpoint 6.4: For scripts and applets, ensure that event handlers are input device-independent.

Whenever the website makes use of the device-dependent onclick java script action, there is a redundant equivalent to carry out the same action. For instance, popup windows will open up in the same window if the `onclick` procedure fails (checkpoint 6.3. describes this in more detail).

Checkpoint 7.3: Until user agents allow users to freeze moving content, avoid movement in pages.

There is no moving content on the website.

Checkpoint 8.1: Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies

The website does not contain applets.

Checkpoint 9.2: Ensure that any element that has its own interface can be operated in a device-independent manner.

Checkpoint 9.3: For scripts, specify logical event handlers rather than device-dependent event handlers.

This has been already described in checkpoint 6.4.

2.2. HTML 4.01 Strict and CSS 2.0

The website after optimization does fully conform to the above stated standards, as explained in more detail in section 2.1.4, checkpoint 3.2.

2.3. Summary

The accessibility and usability of the HC Compact website has been enhanced considerably. The website now conforms to all priority 1 and priority 2 checkpoints of the WCAG. This also includes adherence to the HTML 4.01 Strict and CSS 2.0 formal grammars.

3. Search Engine Optimization and Marketing

3.1. Introduction

Search Engine Marketing (SEM) and Search Engine Optimization (SEO) are sets of methods that pursue the goal of attracting visitors to a website from search engines. The basic difference between SEM and SEO is that SEO forms a subset of SEM.

Search Engine Optimization involves, in particular, refining a website's structure and content so that search engines can crawl it and show links to this website in search results. SEO seeks to produce websites that will be displayed as high as possible in search results for relevant search phrases. The underlying reason for this is to *convert* the visitors, in other words to make them carry out a specific action, such as making a purchase, signing up for a newsletter or viewing contact details. Measuring the success of a SEO campaign often consists of analyzing the number of conversions expressed as profit gained from converted customers.

Unlike SEO, Search Engine Marketing is a broader subject that brings SEO into connection with the overall company's online marketing strategy. It includes techniques as to how to maximize the profit from paid advertisements (displayed as "paid results" in search engines), how to measure conversions of *leads* (people that inform themselves about a product online, possibly on a third-party website, but make the actual purchase offline, e.g. in a brick store) and how to create a budget proposal for a SEM campaign.

This project covers SEO in depth and describes the majority of techniques that SEO embraces. The extent of this work does not allow expanding upon SEM. However, it does touch on the basics that are crucial for proper evaluation of a SEO campaign.

3.2. How search engines work

3.2.1. Crawling and indexing

Search engines consist of several elements. To begin with, they contain a program known as **spider** (sometimes called a **crawler**), which discovers web pages located on the Internet and follows links pointing from them to other pages. The spider ensures that the pages it comes upon will get indexed. Indexing is a process of storing certain data about a web page into the search engine's database. Crawlers should, in theory, be able to find all web pages that are linked to by at least one other page. However, this is not always true, as they often have difficulties following links that are made up solely by JavaScript functions and those that are part of a Flash presentation. Some search engines therefore allow website's developers to manually add a page into their indexing database. Sometimes even sitemaps of entire websites can be submitted, as is the case of Google. On the other hand, there are ways to prevent a spider from indexing a certain page.

The spider continually revisits the websites and keeps the indexing database updated. There are host of variables that the spider takes into account when deciding how often it will visit a given page. Taking the example of Google, it tends to revisit a page the more often the more it values it (using the pagerank as a determiner, as described later on). Also, a page that is found to be often updated is likely to be revisited with a greater frequency.

Put simply, the indexing database contains an index of all words that have been found on the Internet, along with references to websites that contain the given word.

What has been described so far is a continuous task that a search engine conducts in order to keep an updated, simplified and sorted cache of the websites on the Internet. In what follows it will be explained how these data are used to provide a searcher the most relevant search results when he or she actually uses a search facility.

3.2.2. Analyzing the search query

Search query is a term that describes what searchers type into a search engine. It is usually a string that consists of several words, some of which may have special meanings (e.g. wildcards). The words contained in a search query are sometimes called **search terms**. The first job a search engine has to do when a searchers submits a search query is to analyze it.

The exact process of analyzing a query differs among search engines. The following paragraph outlines the basic principles that the majority of search engines draw upon.

The search engine usually attempts to find relevant **word variants** of each search term. A word variant of a term may be for example a plural version of the original term. The search engine may therefore look for “phenomena”, even if the searcher requested “phenomenon”.

Often, search engines allow the user to quote an **exact phrase**, in which case the result must contain all the words in the order specified. Searching for “miserable failure” with quotes and without quotes will probably bring up different results. It should also be mentioned that search engines often look for phrases even if none is explicitly specified. This interrelates with keyword proximity, as explained further.

Search engines often ignore certain terms. These are referred to as **stop words**. For instance, articles (the, a, an) are usually ignored, as they rarely carry some meaning. However, search engines ought to be able to discern situations when these words do bear some information, as might be the case of a search query “The Who” because it is a full name of a rock group.

Usually, search engines offer a set of operators which can be used in conjunction with other words. These include wildcards (* for any word) or modificators like minus if we do not want a particular word to appear in the results.

Once the search query is analyzed, the search engine proceeds to the next stage, which is retrieving relevant pages from the indexing database. This report does not include methods on how this task is implemented, so let us assume that we have already got a set of pages that match the search criteria.

3.2.3. Ranking and sorting the search results

The next step is to sort these pages so that the best ones appear on the top of the results. These algorithms are referred to as **ranking algorithms**. These are complex methods that take into account a multitude of factors, each of which may be of different significance. The primary goal of ranking and sorting search results is to provide the searchers the most relevant source of information (for now, let us ignore paid results). The following factors are used as determiners for assessing the importance of a page for a certain keyword:

Keyword density – The more times the keyword occurs on the page the better. This, however, holds up only to a certain level. Some SEO marketers think that the ratio between a keyword and other text should not exceed 7% (Moran 2006).

There is, however, another aspect of keyword density as well. If a search query contains, say, 3 words, then a search engine may also determine how rare/frequent these words are generally on all pages it has indexed, and decide which one of these three words it should use as a differentiator that will carry more importance. For example, if you search for „kettler exercise bike“, it will probably give more significance to „kettler“, as this is not as common as „bike“ or „exercise“.

Keyword proximity – In the above example, if a page contains „kettler exercise bike“ exactly in this wording, it gives it more significance than to a page where these three words are distant from each other. Again, there is a limit and if it is overlapped, the search engine may interpret this as over-optimization, in which case it will degrade the page's relevance for the given query.

Keyword prominence – It is important in which element the keyword is found. The most important element is the TITLE element. If a search engine finds a keyword it is looking for in a title element, it will probably regard this page as being about that particular word. (Again, it may use linguistic techniques to estimate the correlation between the actual content and the title and determine if the title is relevant indeed or just an attempt to cheat search engines.) Usually, titles are also used in search results along with short extracts (also known as *snippets*). The importance of this element is thus doubled because searchers often decide whether to click on a link based on the wording of the title.

Apart from titles, headings (H1-H6) are the second most important elements that carry most weight. In addition to this, emphasised text is also of importance. Also, some search engines look at the URL for relevant keywords. This is why SEO practitioners often rewrite dynamic URLs by more meaningful equivalents that appear to be static URLs.

It is to point out that metatags like description and keywords are often completely ignored by search engines. This is because many people used these elements to list irrelevant keywords in order to deceive search engines in the past. Search engine therefore look for elements that are displayed to users and find the semantic correlations by their means.

Link popularity – This factor to estimate the importance of a page has been introduced by Google and subsequently borrowed by many other search engines. The idea is to regard other pages linking to the page under consideration as a way of recommendation of the given page. This concept originates from the academic world where referencing a work implies that it has been found a useful, possibly rich source of information.

Google coined its link popularity indicator *pagerank*. The pagerank of a given page is the higher the more external links (also called *back links*) point to the page. It is also the higher the higher the authority of such a linking source is, in other words if a page with pagerank 6 links to another page, it confers to it more authority than a page with pagerank only 3. Another aspect of pagerank is that if a page links to ten pages, it conveys only a tenth of its authority to each of these pages, compared to the case when it links only to one page. To sum it up, it would be ideal to have lots of authoritative pages linking to our page, without them linking to anybody else's page.

It is also to mention that Google assesses the thematic correlation between pages that are interlinked and regards the link the more important the more related the pages are. Also, it looks at the anchor text and especially for Google, the anchor text that is used in an external link carries enormous weight, as this is something that the author often cannot influence to his or her own benefit.

There are actually two pageranks that Google makes use of (if not even more). One is public and one is secret. The public one is given on a scale from 0 to 10; 10 meaning the greatest popularity. The non-public one uses a wider scale to differentiate the number and importance of inbound links. The public pagerank can be computed as a logarithm of its non-public counterpart. The result of this is that most pages have a public pagerank from 0 to 7, whilst only few reach 8 or more. For example, if 20 more links were enough to get from pagerank 3 to pagerank 4, then you would need, say, double that amount to get from pagerank 4 to pagerank 5 (depending on the logarithm base). Note: This is a simplification that seeks a clear demonstration rather than a rigorous mathematical definition. As the author has come to the conclusion that the underlying mathematical formulas for determining pagerank, as published by Henziger (2005), are not necessary to understand in full detail in order to accomplish a successful SEO campaign, it was decided that these definitions will be omitted from this report.

3.2.4. Optimizing for users or for search engines?

The exact implementation of ranking algorithms in search engines is usually proprietary, though many concepts are publicly discussed and brought up at conferences. For example, Google publishes a host of scientific articles on http://portal.acm.org/citation.cfm?doid=1083356.1083357. The public, and notably SEO marketers, are therefore aware of some principles of the ranking algorithms. However, the exact formulas that define the correlation and importance of all ranking factors are kept secret. SEO marketers may, for instance, determine the position of a certain page for a certain keyword in search results, then change the wording of some text on a webpage, wait for a crawler to revisit the page, and subsequently gauge the impact on search results. The drawback of this approach is that the ranking algorithms incessantly change. SEO marketers may therefore highly optimize a webpage today and get it to prominent places in search results but they never know if the tomorrow's ranking algorithms will value the page differently. The results of a Search Engine Optimization are therefore to some degree unpredictable.

Some SEO marketers therefore prefer to obey general principles which they know the majority of search engines value. Others, however, see prominent places in search engines so important that they constantly improve the website’s contents to reflect the current estimated preferences of the search engines. Some go too far in this effort and incorporate dishonest techniques that may temporarily boost their position in search results. However, it is usually a matter of time when a search engine becomes clever enough to disclose such deceptive techniques, in which case the page is usually banned (completely removed from the indexing database).

To sum it up, there are three parties: search engine developers, search engine optimizers and end users of search engines. The first group endeavours to produce such ranking algorithms that best match end user expectations, whereas the second group strives to persuade search engines that it is their page that the user wants to see. If search engines were clever enough to impeccably imitate end-user preferences, this gap between search engine developers and search engine optimizers would not exist, as the only goal of a search engine optimizer would be to produce a page that ideally reflects user expectations. Search engines would only mirror these expectations.

Nevertheless, if we want to rank high in today’s search engines, we have to design websites that are both user friendly and search engine friendly. The following sections demonstrate the concrete steps that are to be undertaken to achieve this.

3.3. Drawbacks of the old website and solutions

3.3.1. Meta elements

Looking at the old `/index.php` file, we can see that the title, description and keyword metatags are the same for all pages:

```
<meta name="description" content="HC Compact - Zdeněk Hrubý (sportovní
   výživa, rotopedy, ergometry, steppery, běžecké pásy, cyklotrenažéry,
kladky,
   posilovací lavičky, sportovní oblečení, cvičební pomůcky)">

<meta name="keywords" content="HC Compact,výživa,sportovní výživa,dietní
   nápoje,iontové nápoje,vitamíny,stimulanty,proteiny,rotopedy,adaptéry,
   ergometry,běžecké pásy,trenažéry,steppery,veslařské trenažéry,posilovací
   lavičky,sport,kladky,Carne Labs,Kettler,Nutrend,Plutino,ATP">

<title>HCC - HC Compact</title>
```

The title element has to be refined to concisely express the contents of the page, e.g. product name in the case of product detail page. Apart from specifying what a page is about the title should also include the name of the company. Stating the company name first may help the company’s branding but may distract a searcher skimming the results from left to right from what he or she was actually looking for. It has been demonstrated that searchers want their search query to appear in search results, in best case exactly in the same wording, in title, in snippet and in the URL (Moran 2006).

Although the description metatag is often ignored when determining page relevancy for a search query, it is sometimes used as a snippet text (or at least its part) in search results. Ideally, this metatag should contain information relevant to the given page and should not be omitted.

3.3.2. Meta elements revised

The new version of the website sets all the three meta elements appropriately across the entire website. This task is done by the `/include/metatags.php` module which exploits the PageClass. The titles contain the main topic of each page, followed by the company name. Taking the example of a page with details about the Kettler Paso 100 exercise bike, the title looks now the following: “Kettler Paso 100 | HC Compact”. The description metatag in this case contains a beginning of the product description. The keyword metatag is created by means of a set of regular expressions that convert the title into a comma-separated string of keywords.

3.3.3. Headings

The old website made use of H1, H2 and H3 elements. The new version does this too, with the difference that H1 is usually used for product or category name and H2 is used for the company name. In the old version this was the other way around. The rationale for this is that the company name does not need to be given such weight, as the website will usually be placed at the first position for search queries that contain its name. On the other hand, search queries like “kettler exercise bike” are much more competitive, and having this phrase in H1 rather than H2 may pay off.

In the course of testing, it was also attempted to use H4 and H5 for category names that appear in the left-hand column. Also, excessive use of H1 elements was tested. In both cases, the positions in Google did not improve. On the contrary, it appears that because of this overuse of heading elements Google penalized one product category that appeared on the first position even before optimization, because it reappeared there shortly after these excessive headings had been removed. Or it may not have been a penalization but an intrinsic consequence of “weight” (expressed in use of headings) being shifted to another keywords.

3.3.4. URLs on the old website

The old website often contained more than three variables in the URL, for example:

```
/index.php?xx=3&zo=&id_detail=1146&hl_detail=250&pod=250&firma=
```

This URL was formerly used to access details about the Kettler Paso 100 exercise bicycle. The problem with such addresses is that some search engines may not index it at all. From the point of a search engine, it is a hard task to determine the nature of dynamic variables. For example, the above link would work the same if the `firma` or `zo` variables were omitted. However, search engines can only make guesses about which parameters are completely redundant, which only change some presentation details, and which do shape the page to a great extent. Some search engines therefore index a page only if it does not contain more than a specific number of parameters.

Some parameters may be even worse than others, such as the PHPSESSID parameter. This is sometimes used to identify user sessions, though from the perspective of a search engine this is a catastrophe because this variable changes each time the crawler attempts to index the page. The crawler either has to employ some methods to determine the nature of dynamic variables or it simply ignores such pages.

3.3.5. Rewriting URLs

Clearly, it is not possible to avoid dynamic pages at all just to make the job easier for search engines. It is, however, possible to set up the web server so that it maps these dynamic URLs into more descriptive ones. As the HC Compact websites exploits PHP in combination with the Apache web server, the Apache module called mod_rewrite has been used for this task.

Mod_rewrite allows a developer to define a set of rules to map a rewritten URL into a real URL by using regular expressions. These rules can be either placed in the `httpd.conf` file of the Apache server, or alternatively in the `.htaccess` file that is located in the root of the given web. This project exploits the `.htaccess` file because the `httpd.conf` is directly inaccessible on public hostings.

Let us look into the .htaccess file of the HC Compact website, on line 89:

```
89 RewriteRule ^(.+-[0-9]+/)*.+-([0-9]+)/([0-9]+)$
index.php?xx=2&pod=$2&page=$3
   [QSA,L]
```

This is a definition of a rewriting rule. The first parameter is a regular expression denoting a set of URLs that will match this rule. If a user requests an URL that matches this rule, the Apache server looks at the second parameter and uses it as the real URL for invoking a PHP script.

If we now request the following URL…

```
/rotopedy-250/rotopedy-kettler-398/1
```

…the Apache server attempts to match this URL to all rewrite rules defined in the `.htaccess` file, starting from the first one. If the first rewrite rule does not match, it proceeds to the next rule, and so on. Once the parser finds a matching rule, it translates it to its real URL equivalent and then either carries on with subsequent rules or stops. Implicitly it carries on and tries to match the rewritten URL with patterns that follow. This behaviour can be suppressed by using the [L] modificator, as shown above. [L] stands for *last*.

For the above URL, the Apache server reaches the last rule, which is shown above as well. Here it assigns “398” to the second parameter, and “1” to the third parameter (both shown in bold in the script excerpts). The resulting URL will be:

```
/index.php?xx=2&pod=398&page=1
```

If the QSA (Query String Append) directive is set, any variables set in the rewritten URL will be copied to the real URL. Thus, if the rewritten URL was for example:

```
/rotopedy-250/rotopedy-kettler-398/1?foo=foo
```

The resulting URL would be:

```
/index.php?xx=2&pod=398&page=1&foo=foo
```

3.3.6. Redirecting old URLs

Once the new website has been created and all links replaced with their rewritten equivalents, the website works fine. If a user bookmarked a page using the old URL it will work as well. There are now two ways how to access one physical script and the user just uses the old one.

However, there is one more thing to tackle: if search engines now start to index our new website with rewritten URLs they will not associate these new URLs with the old URLs. This is a problem because the old pages have probably gained some pagerank already and it would now be lost. Fortunately, there is a way to let search engines know that an URL has been moved to another URL. It is the 301 HTTP header (Moved permanently). Most search engines do understand this header properly and transfer previous ranking from the old URL to the new URL.

For the HC Compact website, a module that caters for redirections has been created and is located at `/include/mod-rewrite.php`.

It is to mention that this module accomplishes one more task: it redirects requests from several alternative domains into only one domain. In our case, the HC Compact website can be accessed not only by the www.hcc.cz domain but also by www.hccbrno.cz, hccbrno.cz, www.hccbrno.com and hccbrno.com. All these domains, however, link to one physical source. It is bad practise to let search engines index more than one domain since it will lead to further pagerank split. Ideally, there is only one domain where the pagerank accumulates.

3.3.7. Google My Sites and Sitemap

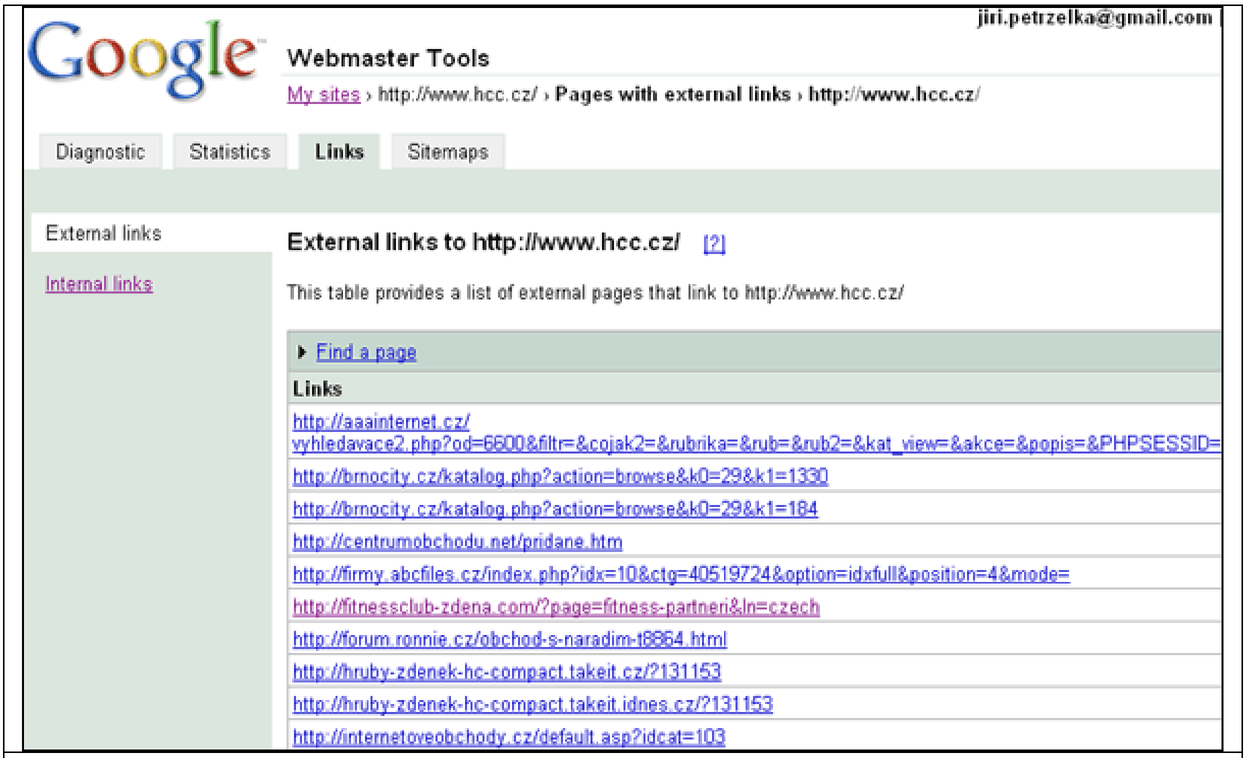
To ensure that all pages of a website will get indexed from Google and also to receive valuable feedback from its crawler (named Googlebot), Google provides a tool called “My Sites”. This utility comes useful if we want to verify that Googlebot can reach pages located on our website. Apart from that, it shows which pages link to our website and which anchor text they use. This comes useful for link building, as described in the following chapters.

The Google My Sites tool can also be used for submitting a Google Sitemap. This is an XML document that contains a listing of all pages located at a given website. The format of this document is specified formally by an XML schema document that is published by Google.

Google then uses this document as a hint as to which pages the website contains. The Sitemap document is usually placed in a directory of the website and Google only needs to be told the URL. Once it knows the location of the sitemap, it will download it regularly and use it as a hint when crawling.

Another advantage of Google Sitemap is that it allows a webmaster to include URLs that contain the results of an internal search facility. These are often URLs that are not directly accessible by following regular links.

For the purpose of this project, a sitemap has been created at the following location: `/sitemap.php`.



External links to http://www.hcc.cz/

This table provides a list of external pages that link to http://www.hcc.cz/

Links
http://aaainternet.cz/ vyhledavace2.php?od=6600&filtr=&cojak2=&rubrika=&rub=&rub2=&kat_view=&akce=&popis=&PHPSESSID=
http://brnocity.cz/katalog.php?action=browse&k0=29&k1=1330
http://brnocity.cz/katalog.php?action=browse&k0=29&k1=184
http://centrumobchodu.net/pridane.htm
http://firmy.abcfiles.cz/index.php?idx=10&ctg=40519724&option=idxfull&position=4&mode=
http://fitnessclub-zdena.com/?page=fitness-partneri&ln=czech
http://forum.ronnie.cz/obchod-s-naradim-t8864.html
http://hruby-zdenek-hc-compact.takeit.cz/?131153
http://hruby-zdenek-hc-compact.takeit.idnes.cz/?131153
http://internetoveobchody.cz/default.asp?idcat=103

Figure 3.3.7. Google My Sites

3.3.8. Back links

Google is known to assign great weight to pagerank and back links. There are several methods to obtain back links.

Probably the simplest one is to submit a **link to catalogues**. Doing this is helpful not only for link building but we can also attract visitors that prefer browsing catalogues rather than using search engines. When adding a link to a catalogue, it is important to list it in a relevant section, as this will help both people and us because a link from a page with a relevant topic is valued more by Google than a link from an unrelated page. Clearly, the intention of Google is to imitate a real user and the usefulness of such links as he or she would perceive it.

When submitting a link to a catalogue, it may be useful to first determine its own pagerank and, based on this, decide whether it is worth the effort. Another important thing to realize is that the main page of a catalogue usually has much higher pagerank than a specific category where our link will be placed. While the pagerank of the main page of the majority of Czech portals varies between 4/10 and 7/10, the actual category will often be no more than 2/10. This is the nature of link-building by using catalogues – it may be quite easy but we rarely get an authoritative link that would boost our own pagerank. Nevertheless, link building has its pluses, especially for new websites that need at least some pagerank to begin with.

Probably the most valued catalogue is the DMOZ (Directory Mozilla), located at www.dmoz.org. This directory claims to be the largest human-edited directory of the web (Netscape 2007). Submission of new links is done by volunteers. A website added to DMOZ must satisfy many requirements, which is the reason why it is difficult to get to DMOZ. However, once a link is included in DMOZ, we can expect a rise of pagerank because many search engines value the DMOZ data highly.

For the purpose of the practical part of this project, the HC Compact website has been submitted to 45 catalogues. These are to be found in Appendix 1. Importantly, the HC Compact website has also been added to DMOZ.

The downside of the catalogues is obvious – the deeper we go into the directory listing, the lower usually its pagerank.

To overcome this, many SEO marketers strive to place their links to other websites. It can be, for instance, a website that is about a similar topic. Ideally, it is a website that does not offer goods for purchase but is a valued source of information relevant to what we sell. We can then make a deal with webmasters of such websites. Either we link to each other reciprocally or we offer them something for placing our link on their website. In the SEM parlance, people that inform themselves using one source and subsequently proceed to shop using another source are referred to as *leads*. The mechanism described can help us attract leads as well as boost our pagerank, especially if the other webmaster is willing to place our link on all his/her pages.

There have been several bilateral agreements as far as the HC Compact website is concerned.

3.3.9. Internal links

Another issue regarding SEO is the management of internal links. As has already been hinted at in the previous chapter, pages that are placed deeper in the directory structure are likely to have a lower pagerank than those placed nearer to the root directory. There are two reasons for this:

a) external links usually point to the home page

b) internal pages usually contain a link pointing back to the home page for simpler navigation

The second reason may seem irrelevant. However, we have to realize that the pagerank algorithm counts internal links within a website as well, despite these links being not regarded as trustworthy as their external counterparts because webmasters have full power over them.

Another detail to note is that both internal and external links should consistently use only one version of possible links to point to one source. For instance, we should choose www.hcc.cz/ and stick to it, rather than using sometimes www.hcc.cz/ and sometimes www.hcc.cz/index.php. The search engines may treat them as two different pages and split their pagerank.

As for the internal links of the HC Compact website, these have been refined to use only one version to point to the home page. In the case of external links, every effort has been made too, though not all links existing prior to the optimization have been revised.

3.3.10. Keywords analysis

Referring back to keyword proximity from chapter 3.2.3., it is crucial that a website contains such words and phrases that the searchers really use. It may be better to use the phrase “exercise bike” rather than “stationery bike” if we come to conclusion than the majority of people use “exercise bike” as a search query. Ignoring the competitors’ website, we would ideally use such phrases that the most searchers use.

However, there are our competitors who also know which keywords and word phrases are to target, as a result of which we have to consider not only the popularity of some phrases but also assess the competitors’ website and estimate the effort needed to optimize our website for a given phrase.

Sometimes, it is better to target a less frequent phrase that is used only by some competitors. We can then get higher in search results and possibly reach at least some visitors. Had we chosen a more competitive phrase and get for example to the thirtieth place in results for a given phrase, there would probably be hardly anyone reaching our link.

The keyword analysis stage often begins with a list of possible keywords that we garner from our own ideas or, better still, from ideas of potential customers. Then, we cross out those keywords that appear to draw only few customers, or customers that are not likely to convert. Note that sometimes we may drive only a few customers to our website who, however, convert very often. The task of the search engine optimizer is therefore to analyze what type of person with what intention is likely to use a given keyword. Buyers usually go through a complex cycle from informing themselves, learning, shopping and finally buying. The search engine optimizer should be aware of customer behaviour and use it as a hint as to which keywords are to target and which not. Unfortunately, due to its length, this report can not cover the psychology of customer behaviour and the conversion cycle in full detail.

Once a website is up and running, the search engine optimizer should analyze the keywords that searchers use to reach the website and based on this data he or she should be continually refining the keywords.

3.3.11. Keywords on the HC Compact website and Google Analytics

The HC Compact website had already been running at the time SEO was launched on it. The author could therefore not only guess which keywords are to target but also determine the real customer demand by analyzing the website’s traffic. For this purpose, the Google Analytics tool has been used. It is a utility developed by Google that allows a search engine optimizer to thoroughly analyze almost all aspects of customer behaviour on a website.

The full potentiality of Google Analytics has been exploited to optimize the website but because of space limit of this report, only certain strategies will be described. In what follows it will demonstrated how Google Analytics proved useful in refining one particular keyword.

There is one category on the HC Compact website that consists of table tennis equipment, namely rackets and tables. The category was initially divided and named as follows:

Tennis tables (Tenisové stoly)

- indoor (vnitřní)
- outdoor (vnější)

The products placed in these categories unanimously contained the ‘tennis table’ phrase in title and in description as well.

Looking at the statistics from January 2007 and visitors coming from organic search, it was found that there were 3 visitors coming though ‘tennis rackets’ (‘tenisové pálky’) and 3 visitors coming through ‘tennis tables’ (‘tenisové stoly’). None of them looked at contact details (none converted into a lead) and none bought anything using the online shop facility.

Such a low traffic appeared to be caused by an unclear distinction of ‘tennis’ from ‘table tennis’. The category names might suggest that the category is about tennis, not about the table tennis.

On February 8, the names were renamed and the category further divided as follows:

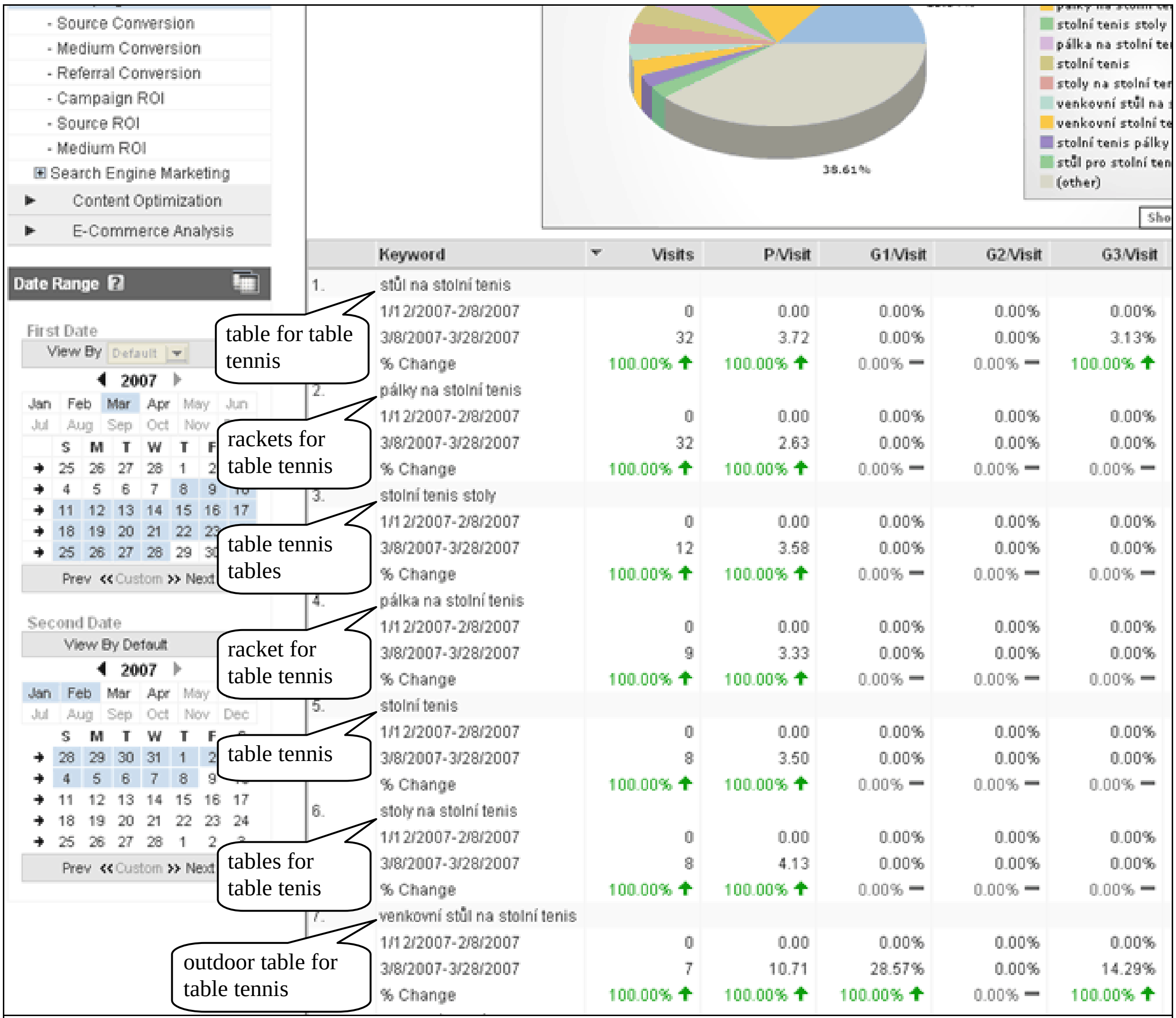
Table tennis (Stolní tenis)

- indoor tables (vnitřní stoly)
- outdoor tables (vnější stoly)
- rackets for table tennis (pálky pro stolní tenis)

Products titles placed in these categories have been renamed as well to contain the ‘table tennis’ phrase.

As a hint, the automatic suggestion tool on Seznam (www.seznam.cz) has been used. Also, the Etarget (www.etarget.cz) has been exploited.

After one month (waiting for spiders to re-index the pages) the results were analyzed. Figure 3.3.11 shows a comparison for organic search results for the periods a) from 12 January 2007 to 8 February 2007 and b) from 8 March 2007 to 28 March 2007. The screenshot contains keywords that have been used by searchers that reached the HC Compact website from all search engines. The keywords are narrowed down to contain the ‘tenis’ expression.



	Keyword	Visits	P/Visit	G1/Visit	G2/Visit	G3/Visit
1.	stůl na stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	32	3.72	0.00%	0.00%	3.13%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	100.00% ↑
2.	pálky na stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	32	2.63	0.00%	0.00%	0.00%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	0.00% –
3.	stolní tenis stoly					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	12	3.58	0.00%	0.00%	0.00%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	0.00% –
4.	pálka na stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	9	3.33	0.00%	0.00%	0.00%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	0.00% –
5.	stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	8	3.50	0.00%	0.00%	0.00%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	0.00% –
6.	stoly na stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	8	4.13	0.00%	0.00%	0.00%
	% Change	100.00% ↑	100.00% ↑	0.00% –	0.00% –	0.00% –
7.	venkovní stůl na stolní tenis					
	1/12/2007-2/8/2007	0	0.00	0.00%	0.00%	0.00%
	3/8/2007-3/28/2007	7	10.71	28.57%	0.00%	14.29%
	% Change	100.00% ↑	100.00% ↑	100.00% ↑	0.00% –	100.00% ↑

Figure 3.3.11. Keyword analysis using Google Analytics

It can be seen that the amount of visitors coming to the website using the ‘tenis’ expression rose considerably. If the screenshot contained all phrases that matched for ‘tenis’, we could see that the number of visitors in the first period was 13, whereas in the latter period it totalled to 202. What is more, the percentage of people visiting the page with contacts of the brick store was 0% and 2.48% for the first and the second period, respectively. Also, there were 1.49% of visitors who did a purchase online in the second period, whereas in the first period there was none.

The above example illustrated how important it is to choose keywords that are commonly used by searchers. The initial conjecture was proven, as it can be seen that searchers do prefer to include ‘table’ into the search query and search for ‘table tennis’, rather than just ‘tennis’.

Similar improvements have been conducted in several other sections.

3.3.12. Copywriting

Copyrighting is usually another part of a SEO campaign. The task of copywriters is to create attractive texts both for customers as well as for search engines. In the former case, this should drive people to conversion actions, while in the latter case the main purpose is to use such keyword combinations so that search engines will regard the page to be a valuable source of information on the given keyword. Sometimes, there is debate whether copywriters should primarily write their copies bearing in mind customers' preferences, or rather create hard to read texts that overly reiterate several keywords.

The practical part of the project did not concentrate on copywriting, as this clearly overlaps its extent.

3.4. Analysis of the results of SEO

3.4.1. Which factors to measure?

The most commonly used method for evaluation SEO results is to compare revenues before and after optimization. Also, we have to bear in mind possible seasonal trends which might blur the results, primarily the Christmas period. In the case of fitness equipment, it is also likely that people will buy more in winter and less in summer, as in summer there are plenty of other sport activities to do. The best thing will be to compare revenues in a given month with the same month previous year.

However, we should also take into consideration that the SEO optimization was in this case done parallel to scores of other design improvements that might have driven customers to buy because of aspects not directly related to SEO, such as changes seeking to adopt good user interface design practices, namely consistency, familiarity, affordance, style and several principles of the Gestalt philosophy (law of symmetry in the case of product boxes, and law of isomorphic correspondence in the case of several new icons incorporated into the new design). Any rise in revenues must therefore be understood in a wider perspective.

Unlike revenues which may be blurred by other factors, we can look at the number of visitors coming through organic search results. This number is clearly dependent to a great extent on SEO, though it has another drawback: it does not say if people coming this way found the website useful, or abandoned it straightaway. To ameliorate this problem we can ignore those people that abandoned the website after seeing the first page. These are clearly people that did not find what they were looking for.

The website has been updated in several steps. The most significant update was conducted on January 10, 2007, when the URL rewriting was launched. Another important period followed in the second and third week of February, when most of the back-linking was done. As for keyword analysis and keyword refinements, these were conducted continually in the course of January-April 2007. The two major search engines that were scrutinized were Google (www.google.com) and Seznam (www.seznam.cz). The meantime between applying changes and these being re-indexed by search engines was usually 7 to 14 days.

3.4.2. Revenues

The graph below compares the revenues from online shopping from December 2005 to March 2006 with the same period next year.

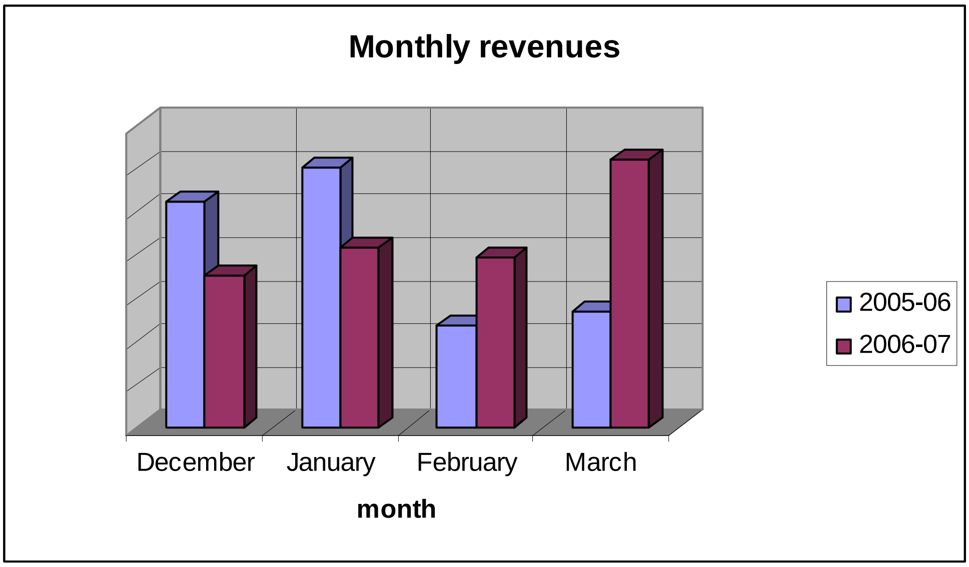


Figure 3.4.2. Monthly revenues

We can see that the revenues were first on decline in December 2006 and January 2007 in comparison to last year's revenues. However, in February 2007, when the Christmas season is over and revenues should therefore drop, the turnover remained almost the same as in January 2007. Subsequently, we can see a sharp rise in sales in March 2007, compared to March 2006. In this period, the revenues more than doubled.

3.4.3. Visitors coming through organic search

First look at the total number of visitors that were directed to the website from all organic search results:

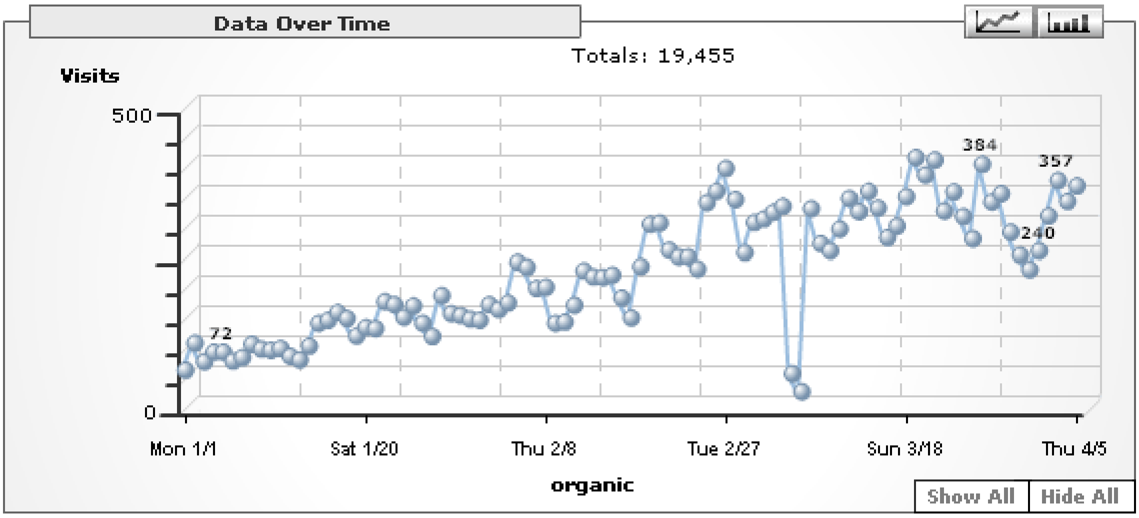


Figure 3.4.3a Visitors from organic search (all search engines)

The sharp deviation on 6 and 7 March was caused by a wrong setting in the Analytics module. The actual number of visits probably correlated with the neighbouring values.

Looking at the diagram, we can see a steady increase of visitors coming from organic search results, averaging 64.1 visitors per day in the first week (1 January to 7 January) and rising up to 284.7 visitors per day in average in the period from 26 March to 1 April. The number of visitors coming though organic search has more quadrupled in the period observed.

In addition to this, Google Analytics can also display the number of visitors coming through a particular search engine. Figures 3.4.3b and 3.4.3c illustrate the number of visitors coming through Seznam and Google, respectively.

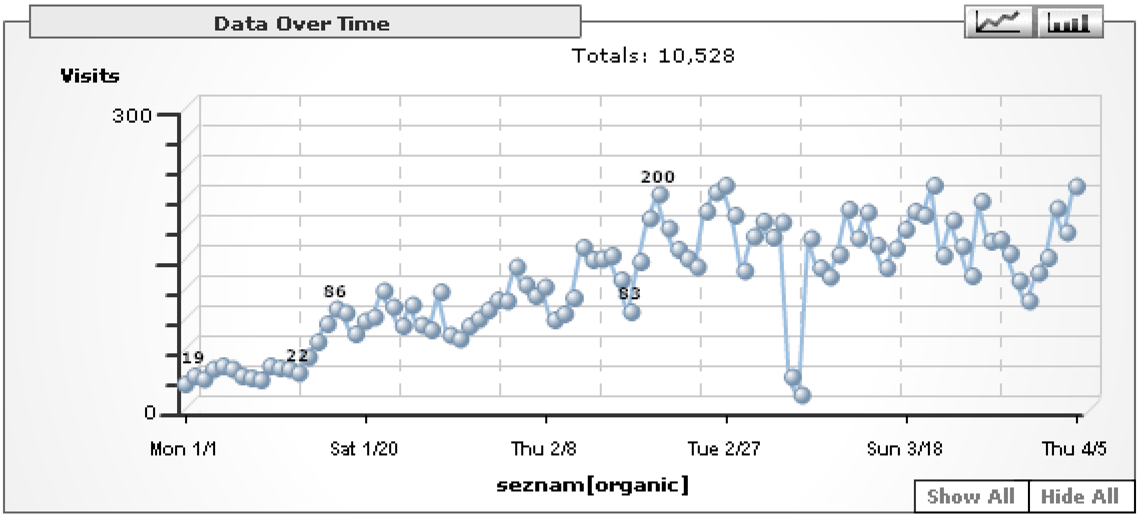


Figure 3.4.3b Visitors coming through Seznam organic search

In the case of Seznam, we can spot two significant moments – first, in the period from 14 January to 17 January, the number of visitors rose from 22 to 86 and since then it did not drop again (e.g. due to differences in customer behaviour during weekdays and weekends). It is very likely that this sharp increase was caused by the rewritten URLs, put in place on January 10. It may be the case that Seznam had difficulties indexing the pages previously because of too many parameters in former URLs.

Another rise is observable in the period from February 17 and February 20, when the number increased from 83 to 200. The odds are that this was caused by Seznam discovering the back links previously submitted to Czech catalogues.

Let us now look at people coming through Google in the same period of time:

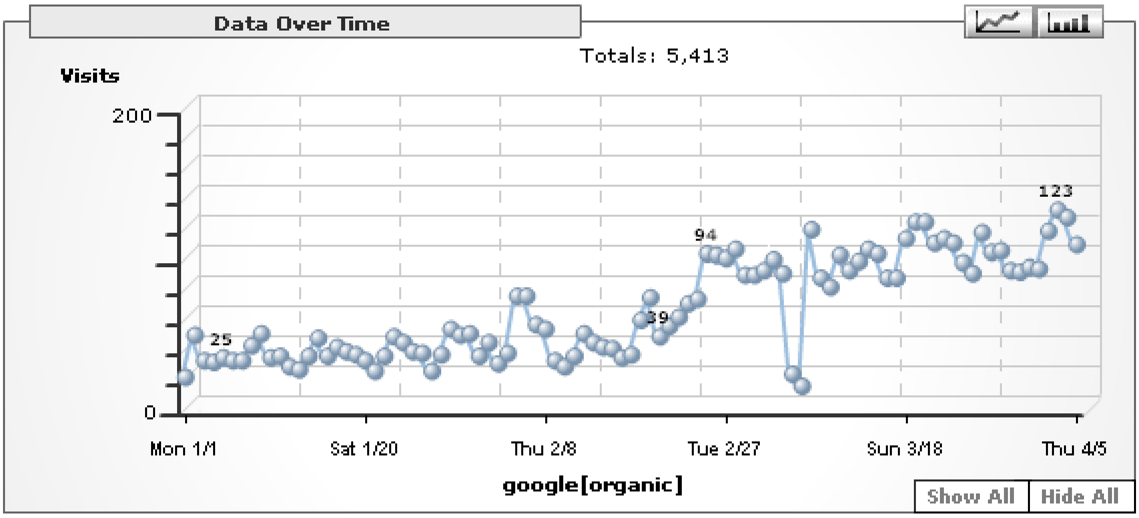


Figure 3.4.3c Visitors coming through Google organic search

Unlike Seznam, Google did not react to the changes put in place on January 10. The cause for this might be that Google had previously had no difficulties indexing former unwieldy URLs.

However, starting from February 20, the number of visitors starts to rise dramatically, ending up at 94 visitors on February 25. There is every likelihood that this resulted from Google having discovered newly submitted backlinks to HC Compact.

3.4.4. Bounce rate

The last thing to consider is the bounce rate. This is the percentage of visitors that abandoned the website immediately after they saw the first page (in the terminology of Google Analytics, they *bounced* upon seeing the first page).

The following diagram displays the total number of daily visitors (averaged through the given week) compared to the number of daily visitors that have not bounced ("real daily visitors"). It can be seen that although the bounce rate increased, the number of real visitors approximately doubled in the course of the search engine optimization.

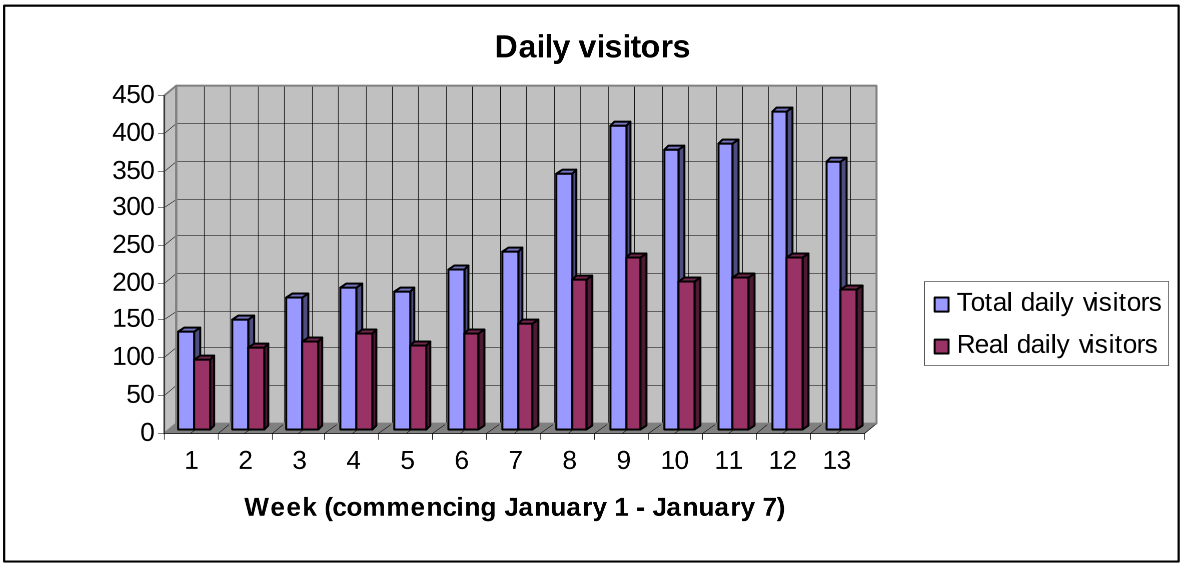


Figure 3.4.4. Daily visitors and the bounce rate

3.4.5. Positions in search engines

Although a position of a particular page for a given keyword in a search engine is a determinant for what has already been illustrated in the statistics above, it still is an important factor that can be used to make certain judgements about the internal implementation of search engine ranking algorithms and also, it can help us in understanding the strategies of our competitors.

There have been about 20 keywords that have been recorded on a regular basis in order to later draw some conclusions about the ranking algorithms that apply at this time (the first quarter of year 2007).

Generally, the most competitive keywords were ‘exercise bike’ (‘rotoped’) and ‘multigyms’ (‘posilovací stroje’). Note that the Czech phrases are now relevant, since the page is Czech and the English equivalents have a different frequency of use by English speaking people.

These two keywords (‘rotoped’ and ‘posilovací stroje’) have been placed into the title of the home page. It was not a trick for search engines, as the home page contains these equipments indeed.

In the case of Seznam, it appears to put an extreme weight to the title meta-element. Whichever keyword was placed into the title element of the home page, the results then brought up this page for this keyword usually among the first five matches. For Seznam, this on-page factor plays a great role, according to these observations. Link building, on the other side, is not as important, though it is one of the ranking factors as well, as demonstrated in section 3.4.3. Seznam uses its own variant of Google’s pagerank which is called S-rank. When the HC Compact’s S-rank was compared with its competitors’ S-rank, the HC Compact website did quite well: Most competitors, like www.sedlakkokes.cz or www.4fit.cz, that score much better in Google for ‘rotopedy’ and ‘posilovací stroje’ have the S-rank of about 50/100, which is very close to the HC Compact’s S-rank (43/100 on 6

April 2007). To sum it up, to optimize for Seznam proved to be not difficult, provided the usual SEO techniques are put in place.

On the other hand, to optimize for highly competitive keywords in Google proved to be more difficult than expected. When first measured on February 21, 2007 the ‘rotopedy’ keyword brought up the HC Compact link on the 28th position. Then the position kept improving until March 18, 2007 when it reached the 16th position. Subsequently, however, the particular page completely disappeared from results and another page from HC Compact website was shown instead, unfortunately on the 23rd position (March 29, 2007). There were several assumptions as to why the former page which ranked 7 positions higher had disappeared but none proved to be correct. It appears that something has changed inside the ranking algorithm of Google. Also, the pagerank does not seem to grow in spite of the link building process. Throughout the whole optimization process, the pagerank of the home page was 3/10. It should, however, be noted, that the pagerank that Google publishes is something else than the internal pagerank. The internal pagerank changes continually and determines the search results, whereas the public version of pagerank is updated only once a couple of months to reflect the real pagerank that is non-public.

Although this project did not succeed in getting the HC Compact website to the first 10 positions in Google for the most competitive words, it still increased the number of people coming from Google organic search, as demonstrated in chapter 3.4.3. Notably, these people come through less competitive phrases. The author of this report surmises, after viewing the websites of HC Compact’s competitors which rank higher, that the clue for success after all lies in copywriting and links from authoritative and thematically relevant sources. This assumption is based upon the increase of visitors coming through Google at the time when Google tracked new back links. Also, many competitors’ websites are interlinked with other relevant websites that have a high pagerank. However, to strike deals with important vendors that have highly valued websites is out of the scope of this project.

Discussion

In the first part, it was demonstrated how the MVC concept can be used in an existing PHP project to increase the updatability and robustness of a large-scale application. The separation of application logic from its presentation proved to have many advantages when conducting the front-end optimization and the SEO optimization, namely it makes programmer's errors less likely, it speeds up the programming and also, adopting the OOP and MVC principles makes it possible for a team of programmers to split their work better in future. The downside of the HC Compact website is that it is too extensive to completely redesign its internal structure. This is, however, a necessary step that will have to be undertaken in future, as the website grows in extent.

The second stage (WCAG compliance) proved to be less complex than the first stage. The priority 1 and priority 2 checkpoints have been observed by making rather small improvements. This is possible because of the first stage; otherwise even these minor fixes would have been hard to do consistently and effectively.

There occurred several points where the WAI instructions were not rigorously specified, as is the case of enough contrast. This could be specified mathematically by a formula, stating the minimal difference in intensity of two neighbouring pixels. Also, in the case of sitemaps, it does not say to which level of detail a sitemap must go. This part of the project, in fact, lacks any mechanisms that would measure the actual merit of adopting the WCAG for this particular website. The standards have been applied but a user testing would be necessary to ascertain that it actually led to tangible improvements. The author is aware of this drawback but due to the extent of the project had to omit this.

The third part is examined in the greatest detail. The reason for this is that the main purpose of any commercial application is to generate profit. Had the website not been profitable, this project could not have been undertaken at all. The third part of the project therefore includes some background of SEO, as well as description of the improvements taken as well as a thorough analysis of results. The SEO optimization resulted in twice as many visitors coming to the website (compared January 2007 and March 2007) and in March 2007 its revenues more than doubled, compared to March 2006. It was shown that SEO improvements have a direct influence on the number of people visiting the website and making a purchase on it.

Nevertheless, the author of this report is convinced that there is still much to do in terms of SEO and SEM. The results in Google clearly indicate that the HC Compact has not beaten its competitors. The author supposes that the root cause for this is an insufficiently rich content (category descriptions and descriptions of some products as well) that especially Google values greatly. After all, Google's primary goal is to find the richest and most accurate source of information for searchers. This information should be supplied by experts on fitness equipment which the author of this report is not. Also, extensive link building is only possible if the SEO campaign is intertwined with the entire company's marketing strategy, and notably SEM. Clearly, a cooperation of several people is needed to create a group of search engine optimizers who will eventually get the page to prominent places in

Google. The author regrets that most attempts to give advice and cooperate with the website owners were either not possible or ignored.

Conclusion

The principal outcomes of this project are: A more secure back-end code exploiting OOP PHP 5 and the MVC architecture, an accessible and usable front-end adhering to priority 1 and 2 checkpoints of the WCAG and a more competitive website that addresses most of the latest SEO techniques.

The project stretches from the design of a modern Content Management System and its programming underlying, to the parts that directly interact with end users, thus forming a balanced work where technical and commercial aspects are seen as inseparable counterparts.

The website could be further optimized in two other ways:

a) optimize the speed of back-end scripts and the size of output

b) observe (directly or indirectly) customer behaviour on the website and improve those parts of the user interface that would be found user-unfriendly.

Also, a more rigorous research could be done into how modern search engines are implemented. This could possibly shed more light on further improvements in terms of SEO.

Bibliography

ASAP Consulting s.r.o. *Dostupné nástroje pro Vaši potřebu* [online]. [2007] [cit. 2007-07-18]. Dostupný z WWW: <http://www.i-asap.net/nastroje.php>.

CERN. *Welcome to info.cern.ch* [online]. c2005 [cit. 2007-07-18]. Dostupný z WWW: <http://info.cern.ch/>.

CUTTS, M. *Archive for Google/SEO* [online]. 2007 [cit. 2007-07-18]. Dostupný z WWW: <http://www.mattcutts.com/blog/type/googleseo/>.

Etarget. *Etarget nástroje : Hledání kombinací* [online]. [2007] [cit. 2007-07-18]. Dostupný z WWW: <http://www.etarget.cz/customer/info/stats.php?cmb=1>.

Google. *Google Analytics* [online]. 2007 [cit. 2007-07-18]. Dostupný z WWW: <http://www.google.com/analytics/>.

Google. *Google Labs : Research Publications* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <http://labs.google.com/papers.html>.

Google. *Google Webmaster Central* [online]. 2006 [cit. 2007-07-19]. Dostupný z WWW: <http://www.google.com/intl/cs/webmasters/>.

Google. *Google Webmaster Tools* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <https://www.google.com/webmasters/tools/siteoverview?hl=en>.

Google. *Using the Sitemap Protocol* [online]. 2006 [cit. 2007-07-19]. Dostupný z WWW: <https://www.google.com/webmasters/tools/docs/en/protocol.html>.

Google. *Webmaster Guidelines* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <http://www.google.com/support/webmasters/bin/answer.py?answer=35769>.

Googlerankings.com. *Google Rankings : Basic SEO advice* [online]. 2003-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://googlerankings.com/basic.php>.

HENZIGER, Monika. Hyperlink analysis on the world wide web. In *Conference on Hypertext and Hypermedia - Proceedings of the sixteenth ACM conference on Hypertext and hypermedia*. [s.l.] : [s.n.], 2005. s. 1-3. Dostupný z WWW: <http://portal.acm.org/citation.cfm?doid=1083356.1083357>.

JANOVSKÝ, Dušan. *Google PageRank* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <http://www.jakpsatweb.cz/seo/pagerank.html>.

KARBAN, R. *České a slovenské katalogy s užitkem pro SEO* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <http://www.seo-expert.cz/ceske-a-slovenske-katalogy-s-uzitkem-pro-seo>.

LANGRIDGE, Stuart. *DHTML Utopia: Modern Web Design : Using JavaScript & DOM*. 1st edition. United States of America : Sitepoint, 2005. 318 s. ISBN 0-9579218-9-6.

MORAN, Mike, HUNT, Bill. *Search Engine Marketing, Inc. : Driving Search Traffic to Your Company's Web Site*. 1st edition. Stoughton, Massachusetts, USA : IBM Press, 2006. 560 s. ISBN 0131852922.

Netscape. *Dmoz - open directory* [online]. 1998-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://www.dmoz.org/>.

PROKOP, M. *SEO FAQ* [online]. 2004-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://vyhledavace.info/seo-faq/>.

ROGERS, Ian. *Page Rank Explained : The Google Pagerank Algorithm and How It Works* [online]. [2007] [cit. 2007-07-19]. Dostupný z WWW: <http://www.iprcom.com/papers/pagerank/>.

SEO Tools : Future PageRank [online]. 2003-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://www.seochat.com/seo-tools/future-pagerank/>.

Seo.unas.cz. *SEO asistent* [online]. 2003-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://seo.unas.cz/>.

SCHLOSSNAGLE, George. *Advanced PHP Programming*. [s.l.] : [s.n.], 2004. 672 s. ISBN 0672325616.

THATCHER, Jim, et al. *Constructing Accessible Web Sites*. 1st edition. United States of America : Springer-Verlag, 2002. 415 s. ISBN 1-59059-148-8.

The PHP Group. *Chapter 29. Using Register Globals* [online]. 2001-2007 [cit. 2007-07-19]. Dostupný z WWW: <http://uk2.php.net/register_globals>.

TURCSANYI, T. *mod_rewrite: A Beginner's Guide to URL Rewriting* [online]. 2002 [cit. 2007-07-19]. Dostupný z WWW: <http://www.sitepoint.com/article/guide-url-rewriting/1>.

W3C. 1999a. *Web Content Accessibility Guidelines 1.0* [online]. 1999 [cit. 2007-07-19]. Dostupný z WWW: <http://www.w3.org/TR/WAI-WEBCONTENT/>.

W3C. 1999b. *HTML 4.01 Specification, Chapter 11 – Tables* [online]. 1999 [cit. 2007-07-19]. Dostupný z WWW: <http://www.w3.org/TR/html4/struct/tables.html#table-directionality>.

W3C. *Core Techniques for Web Content Accessibility Guidelines 1.0* [online]. 2000 [cit. 2007-07-19]. Dostupný z WWW: <http://www.w3.org/TR/WCAG10-CORE-TECHS/>.

Wikipedia Foundation, Inc. 2006a. *Model-view-controller* [online]. 2006 [cit. 2007-07-19]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Model-view-controller>.

Wikipedia Foundation, Inc. 2006b. *Image:ModelViewControllerDiagram.png* [online]. 2006 [cit. 2007-07-19]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Image:ModelViewControllerDiagram.png>.

Wikipedia Foundation, Inc. *Code refactoring* [online]. 2007 [cit. 2007-07-19]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Refactoring>.

Appendices

1 Catalogues where link inclusion to www.hcc.cz has been requested

www.aaainternet.cz
portal.abcfiles.cz
alfa.elchron.cz
www.allytrade.cz/Refer.asp
www.atila.cz
www.bezvaportal.cz
www.caramba.cz
www.cent.cz
www.citysearch.cz/
www.divoch.cz
elipsa.cz
www.infotip.cz/
www.infoweb.cz
jahho.net/
jednorozec.cz/
klikni.idnes.cz/
linkovnik.wz.cz/
www.lukyn.com/katalog.php
www.najduvse.cz
www.odskok.cz/o_index.php
www.opendir.cz
www.czprima.cz
www.o2active.cz
www.vokno.cz/index.asp
www.vsichni.cz
katalog.pcsvet.cz
www.zacatek.cz
www.zdroj.cz
www.rejstrik.net
reklama.euweb.cz
www.cykloserver.cz
www.pingpong.cz
vivat.cz/aa/index.php
katalog.celostnimedicina.cz
www.centrumobchodu.net
www.iobchody.com
www.ishopy.com
www.shopfinder.cz
www.stopa.cz
sportovni-potreby.internetoveobchody.com
www.internetoveobchody.cz
www.topobchody.cz
www.internet-obchody.cz
www.jaknaweb.com
www.cviceni.org

2 Vocabulary

This brief listing may help you understand some interrelations in cases when Czech names had to be preserved.

Czech	**English**
Akce	Special offers
Firma	Company
Mapa stránek	Sitemap
O firmě	About the company
Podrobné vyhledávání	Advanced search
Sledovat změny	Track progress
Sortiment	Products
Vyhledávání	Search

3 English translation of the website

The original, Czech only, website located at www.hcc.cz has been translated into English for the purposes of evaluation and sample data has been created. The English version is enclosed on CD and it has also been uploaded at www.artokna.com/hcc-en/. Since this is a third party hosting the author cannot guarantee that this link will work 24/7. You may use this URL or install the website on localhost, in which case follow the instructions listed in Apendix 4.

The original site is in Czech only. The names of the files and tables in the database are sometimes English and sometimes Czech. The rewritten URLs are always Czech.

In the English translation of the website, all content of the website is English. However, file names and database tables are sometimes Czech and sometimes English. As for rewritten URLs, these are Czech if the name is derived from a directory name that is Czech as well (e.g. /sortiment/ contains the products because there is actually a physical directory called ‘sortiment’). On the other hand, if the URL is derived from English data coming from the database, then the URL mirrors the English version (e.g. /exercise-bikes-250/1 because the category name in the English translation is ‘exercise bikes’). Last thing to point out is that e-mails that are automatically sent upon order placement event and other events are Czech in both versions. This is because to understand this project you do not need to understand the text in automatically generated e-mails.

The seemingly incongruent translation where something is English and something Czech has a rationale: The reader is primarily expected to use the English translation, especially for examination of back-end and front-end improvements, except of SEO. In the SEO stage, however, the translation must not diverge from the original too much because the examiner will probably look at how search engines treat the real HC Compact website. For example, using the ‘hcc site:www.hcc.cz’ statement in Google to determine all the pages indexed by it from the www.hcc.cz domain, it will bring up addresses that contain original Czech names. The translation of URL is done so that you can always pick the part following server name specification (e.g. /rotopedy-250/1 from the original site) and use it in the translated version. Even if you use the Czech version of a category name like ‘/rotopedy-250/1’ and use this chunk of URL in the English version (www.artokna.com/hcc-en/rotopedy-250/1), the English website will automatically redirect this address (www.artokna.com/hcc-en/exercise-bikes-250/1). You can therefore always cross-check the English version, the Czech version and what search engines have indexed.

To test the website, you may register as a new user or use an existing account with the username ‘test’ and password ‘test’. There are sample products in these categories: ‘Exercise Bikes’, ‘Exercise Equipment’ and ‘Medic-Line’.

4 Installation of PHP+MySQL+Apache

The HC Compact website requires the following settings:

- PHP at least version 5.0
    - php.ini settings:
        - error_reporting = E_ALL & ~E_NOTICE;
- MySQL server at least version 5.0
    - A new database must be created and then the `/sql/hccen.sql` script run upon it (you may need to replace line 13 in this script with another database name).
- Apache server at least version 1.3.37,
    - `httpd.conf` settings:
        - `mod_rewrite` loaded and `.htaccess` file enabled for the testing directory
        - `DirectoryIndex` must contain ‘index.php’, not only ‘index.html’ (default)
    - the `.htaccess` file in the root of the project directory may need to be another `RewriteBase`
- `/include/global.php` in the project directory:
    - On line 21 the `$serverDir` variable must be set to `/` if the project runs in root on localhost, or to another directory if the project is located in a directory (e.g. if the project root is `http://localhost/hccen/` then the `$serverDir` variable should be `/hccen/`)
- `/classes/generic/mysql.generic.php` in the project directory:
    - The `$dbhost`, `$dbname`, `$user` and `$password` variables of the MysqlClassArtokna class (lines 8-11) must be specified according to your database settings,
- /admin/inc/ad_mysql.class.php in the project directory:
    - On lines 222 to 225 the database access details must be entered once more.